PATENT
B208-685

EXPRESS MAIL LABEL NO. EL030870561US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants : Fumiaki Takahashi, et al.

Serial No. : Unassigned (Parent 08/942,729 filed 10/2/97)

For : SYSTEM CONTROL METHOD AND SYSTEM CONTROL APPARATUS

Filed : January 29, 1999

Examiner : Unassigned (Parent - L. Lu)

Art Unit : Unassigned (Parent - 2756)

Assistant Commissioner
for Patents
Washington, D.C. 20231

Sir:

CLAIM FOR PRIORITY

Claim is made under 35 USC § 119 for the benefit of the filing date of Japanese Patent Application No. Hei 05-189633 (filed 7/30/93); Hei 05-189634 (filed 7/30/93); Hei 05-189636 (filed 7/30/93) and Hei 05-189637 (filed 7/30/93); copies of which have been filed in prior application Serial No. 08/281,355.

Respectfully submitted,

John J. Torrente

ROBIN, BLECKER & DALEY
330 Madison Avenue
New York, New York 10017
(212) 682-9640
January 29, 1999

John J. Torrente
Registration No. 26,359
An Attorney of Record

24:04:05

16

PATENT OFFICE

JAPANESE GOVERNMENT

This is to certify that the annexed is a true copy of the following application as filed with this office.

Date of Application: July 30, 1993

Application Number: Patent Application No. Hei 05-189634

Applicant(s): Canon Kabushiki Kaisha

August 26, 1994

Commisioner,

Patent Office Akira Takashima

Issuance No.Hei 06-3043522

(Name of Document) Application for Patent

(File Number) 2589046

(Date of Submission) July 30, 1993

(Addressee) To Commissioner, Patnet Office,
Wataru Asoh

(International Patent Classification) H04N 5/232

(Title of Invention) CONTROL SYSTEM FOR MULTIMEDIA DEVICES

(Number of Claims) 13

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Takashi Aizawa

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Akira Mamiya

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Fumiaki Takahashi

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Akira Suga

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Hideaki Kawamura

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Kenji Hatori

(Applicant for Patent)
(Identification Number) 000001007
(Postal Code) 146
(Address) 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Canon Kabushiki Kaisha
(Representative) Hajime Mitarai
(Telephone Number) 03-3758-2111

(Agent)
(Identification Number) 100069877

24.04.95

(Postal Code) 146
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Patent Attorney)
(Name) Giichi Marushima
(Telephone Number) 03-3758-2111

(Indication of Fee)
(Method of Payment) Deposit
(Deposit Account Number) 011224
(Amount of Payment) 21000

(List of Submitted Articles)
(Name of Article) Specification 1
(Name of Article) Drawings 1
(Name of Article) Written Abstract 1
(General Power of Attorney Number) 9003707

(Necessity of Proof) Necessary

(Name of Document)

Specification

(Title of the Invention)

CONTROL SYSTEM FOR MULTIMEDIA DEVICES

(Scope of Claim for Patent)

(Claim 1)

A control system for multimedia devices, comprising a multimedia controller connected via a communication line to said multimedia devices to control unitary management of said multimedia devices,

wherein said multimedia devices are provided with first interface control units arranged on communication with said multimedia controller to function unitarily, whereby said first interface control unit is provided with an interface controller for controlling communication either physically or in the low logic level and a first system controller for communicating with a main function in said multimedia device and for controlling said first interface control unit; and

wherein said multimedia controller is provided with a second interface control unit for controlling communication with all said multimedia devices, wherein said second interface control unit is provided with an

interface controller of the character described above, means for processing the output of said interface controller in the form of signals informing the detection of when said multimedia devices are connected to, and disconnected from, a network, means storing a network monitoring table to be used in said signal processing means, as it is looked up and updated, said signal processing means, and a second system controller for communicating with a main function unit of said multimedia controller and for controlling said second interface control unit.

(Claim 2)

A control system for multimedia devices according to claim 1, wherein said multimedia controller is provided with a first timer circuit arranged on rising of a power source for said multimedia controller to synchronously start operating multimedia, and means responsive to occurrence of an output of said first timer circuit for regularly recycling the sending of a link checking message to said multimedia devices, whereby responsive to advent of said link checking message, said multimedia devices give off the one of said signals which informs of their current connections in the network, and wherein said multimedia controller is further provided with means for updating said network monitoring table based on the information about any changes in the state of the network as obtained from said signal of the current

connections of said multimedia devices.

(Claim 3)

A control system for multimedia devices according to claim 2, wherein said multimedia controller is further provided with resending means operating in such a manner that, if, as all current connections of the network are being checked by said message sending means, the connection informing signals returned from the network indicate that, of said multimedia devices which have already been recorded in said monitoring table, there is a one missed, another one of said link checking message is sent to the one of said multimedia devices which is in question, and erasing means operating in such a manner that when the operation of said resending means results in receiving no acknowledgment from said multimedia device in question, the corresponding entry to said multimedia device in question is deleted out of said monitoring table.

(Claim 4)

A control system for multimedia devices according to claim 3, wherein said multimedia devices are provided with second timer circuits arranged on rising of said power sources for said multimedia devices to synchronously start operating and means responsive to occurrence of outputs of said second timer circuits for regularly recycling the sending of said current connection

informing signals to said multimedia controller, and said multimedia controller is provided with means for updating said monitoring table based on the information about any changes in the status of said multimedia devices as obtained from said current connection informing signals.

(Claim 5)

A control system for multimedia devices according to claim 4, further including means operating in such a manner that, as said multimedia controller is checking the status of individual links of the network by receiving said connection informing signals successively from all said multimedia devices, each time said multimedia controller receives said connection informing signal from the one of said multimedia devices which is not yet recorded in said monitoring table, a counter relevant to said new multimedia device in one-to-one basis is put into said monitoring table, or each time it receives said connection informing signal from the one of said multimedia devices which has already been recorded in said monitoring table, the corresponding counter to said already recorded multimedia device is reset; means for incrementing said counters independently of one another in synchronism with the output of said first timer circuit; resending means responsive to attainment of the counted value of any one of said counters to a threshold for resending at least another one of said link checking message to the corresponding

one of said multimedia devices; and erasing means responsive to detection of a fact that the operation of said resending means results in receiving no acknowledgment from said multimedia device in question for deleting the corresponding entry to said multimedia device in question out of said monitoring table.

(Claim 6)

A control system for multimedia devices according to claim 1, wherein said multimedia devices each are provided with at least two terminals for connection in routes of in the network, circuits for detecting occurrence of route connection in one to one correspondence to said terminals, and means receptive of the output of said route connection detecting circuits and responsive to detection of any changes in the status of links between connection and disconnection for sending either said connection informing signal or said disconnection informing signal to said multimedia controller, whereby with selection of one of said multimedia devices, say the first, for addition of a new or second multimedia device, when connection of said second multimedia device occurs, said route connection detecting circuit in said first multimedia device changes its output, and said first multimedia device responsive to this sends said connection informing signal to said multimedia controller, and then when disconnection of said second multimedia device from said first multimedia

device, said route connection detecting circuit in said first multimedia device changes its output and said first multimedia device responsive to this sends said disconnection informing signal to said multimedia controller, and wherein said multimedia controller is provided with means for receiving said connection informing signal or said disconnection informing signal to update said monitoring table to be used in said signal processing means.

(Claim 7)

A control system for multimedia devices according to claim 6 further including resending means operating in such a manner that when said multimedia controller receives said disconnection informing signal, another one of said link checking message is resent to the one of said multimedia devices which has transmitted said disconnection informing signal and erasing means operating in such a manner that when the operation of said resending means results in receiving no acknowledgment from said multimedia device of destination, the corresponding entry to the removed one of said multimedia devices is deleted out of said monitoring table.

(Claim 8)

A control system for multimedia devices according to claim 1, further including a repeater between

said multimedia controller and a group of said multimedia devices, and wherein said repeater is provided with a corresponding number of terminals for connection in routes of the network to the number of said multimedia devices, route connection detecting circuits in one to one correspondence with said terminals, means responsive to change of the output of any one of said route connection detecting circuits for sending said connection informing signal or said disconnection informing signal to said multimedia controller, an interface controller of the character described before and means for controlling all of these means, and said multimedia controller is provided with means receptive of said connection informing signal or said disconnection informing signal for updating said monitoring table.

(Claim 9)

A control system for multimedia devices according to claim 8, wherein said multimedia controller is provided for said means receptive of said connection informing signal or said disconnection informing signal for updating said monitoring table with resending means for transmitting another one of said link checking message to the one of said multimedia devices which has becomes a cause of giving off said connection informing signal or said disconnection informing signal and erasing means operating in such a manner that when the operation of said resending means results in receiving no

acknowledgment from said multimedia device in question, the corresponding entry to said multimedia device in question is deleted out of said monitoring table.

(Claim 10)

A control system for multimedia devices according to claim (2) or (3), wherein said multimedia devices each are provided with terminals for connection in routes of the network, circuits for detecting occurrence of said route connections arranged in one to one correspondence to said terminals and means receptive of the outputs of said detecting circuits and responsive to any changes in the status of links between connection and disconnection for sending said connection informing signal or said disconnection informing signal to said multimedia controller, and wherein said multimedia controller is provided for said signal processing means for receiving said connection informing signal or said disconnection informing signal with means for updating monitoring table.

(Claim 11)

A control system for multimedia devices according to any of claims (6) to (10), wherein said multimedia devices, said multimedia controller and said repeater have their terminals for route connections provided with switches arranged on insertion of line cables to turn on, mechanisms for locking said line

cables, and actuator members arranged to simultaneously perform releasing of said line cables from the locked state and turning off of said switches, whereby the output signals of said switches are transferred to said route connection detecting circuits.

(Claim 12)

A control system for multimedia devices according to any of claims (1) to (11), further including a first power circuit for supplying electric power to said multimedia controller and said first and said second interface control units of each of said multimedia devices and a second power circuit for supplying electric power from a different system from that for said first power circuit, said second power circuit being arranged to turn on and off either when the user operates a main switch for said multimedia controller or any of said multimedia devices or by messages from said multimedia controller.

(Claim 13)

A control system for multimedia devices according to any of claims (1) to (12), further including means operating in such a manner that at a time when said multimedia controller has updated the entries in said monitoring table to reflect any changes in the state of the network sensed by receiving said connection informing signal or said disconnection informing signal, unique names for identifying said multimedia devices are

allocated in one to one correspondence; and means for sending information of said allocated identification names to said respective multimedia devices, and wherein said multimedia devices each are provided with means for displaying the given one of said identification names.

(Detailed Description of the Invention)

(0001)

(Field of Utility on Industry)

This invention relates to systems comprising a controller for multimedia devices and, more particularly, to a control system for multimedia devices in which the multimedia controller keeps monitoring the status of individual links in the network and checks whether the power sources of the multimedia devices are turned on or off.

(0002)

(Prior Art)

Recently, in many fields of art, beginning with AV instruments such VTRs and CDs, application of digital technologies has started to prevail. Such instruments, when digitized, are taken as so-called multimedia devices to meet a demand for permitting use of pictures, sound, text and other items of information in an integrated format. In the past, for such instruments to connect to each other, the user had to keep their track under his or her own responsibility.

To assure unitary handling of all items of information, on the other hand, it becomes necessary to use an equipment for controlling coordination of these instruments.

(0003)

As a system that well resembles this, mention may be made of the network control by which computers or like information processing devices and printers or other peripheral devices are selectively connected to a network. Even in the control system for multimedia devices, for the purpose of accurately managing and efficiently maintaining the links in the network, there is need to accurately and reliably check the status of individual links by the information processing device (hereinafter referred to as "multimedia controller"). The following checking methods are available.

(1) Every time a device is connected to the system, the user reads the data that identifies the current connection and then manually puts it into a memory for the multimedia controller. When the power source is later turned on, the multimedia controller checks any changes in the status of the network by reading the updated memory.

(2) Every time a device is connected to the system, the user reads the data that identifies that device and then manually puts it into a memory in this device. When the power source of the device is later turned on, the multimedia controller reads it by communication to check

establishment of a new connection in the network.

(3) For every new connection or disconnection in the network, the multimedia controller, when actuated by the user, surveys all current connections and updates the memories in all the powered-on devices by assigning data for identifying the connections from one another in the network to these devices.

(0004)

(Subjects the invention is to Solve)

Since the prior known methods are, however, of technology for computer systems, because some skill and appreciation on the computer system are prerequisite, a problem arises that such habits are hardly infiltrated to ordinary users of VTRs, CD players, or like other home appliances.

(0005)

For example, in the prior art by the method (1) described above, it is by hand that information of all the devices connected to the system had to be previously inputted into the multimedia controller. To identify what data had been assigned to which devices, an authorized person who had done it needed to memorize the assigning table. From this reason, the authorized person must update the entries in the table accurately, so it was unavoidable that malfunctions or the like often occurred due to the inputting of wrong data. If one of the users wanted to connect a new device to the system,

he had to ask the authorized person for making determination of a data for identification and for inputting it into the multimedia controller. This was troublesome.

(0006)

Next, for the method (2) described above, the task of updating the entries in the assigning table is left to the multimedia controller. But it is still by hand that the data for identification are previously inputted into the respective devices which are to be connected to the system. Therefore it was impossible to completely avoid occurrence of malfunctions due to the input mistake.

(0007)

The methods (1) and (2) both are unable to detect any changes in the status of the network, if a new connection or disconnection occurs at a time during the operation of the system. It was, therefore, impossible to accurately recognize what device had been added or removed until the system was once shut down and raised up again.

(0008)

Also, in the prior art by the method (3), the problem described just above are considerably solved. Even if a new connection is added, or one of the current connections is removed, under the condition that the system remains in the rising up state, it is possible to detect this change in the status of the network without

having to once shut the system down and raise it up again. But to update the entries in the assigning table, the user has to do a task of giving a command for checking the status of the network to the multimedia controller. Therefore, the user is left to be always nervous of the necessity of giving that command after a new device has been added to the system, or any of the connected devices has been removed. This was not comfortable.

(0009)

Because of the problems described above, the conventional system had a subject that the ordinary users who acquired no skill and no appreciation on the computer systems could hardly accept it.

(0010)

(Means for Solving the Subject)

With the foregoing in mind, the present invention is made and its object is to solve such subjects as described above. One of the features which are characteristic of the invention is that in application to the multimedia system comprising a number of multimedia devices and a multimedia controller connected to the multimedia devices by a communication network to unitarily manage the multimedia devices, the multimedia devices each are provided with a first interface control unit which performs the unitary function on communication with the multimedia controller, wherein the aforesaid first interface control unit is provided with an interface controller for controlling the communication either

physically or in low logic level and with a first system controller for communicating with the main function unit of the multimedia device and for controlling the first interface control unit; and the multimedia controller is provided with a second interface control unit for controlling communication with each of the multimedia devices, wherein the aforesaid second interface control unit is provided with an interface controller of the character described above, means for processing either a signal informing of the connection of a multimedia device sensed by the interface controller or a signal informing of the disconnection of one of the multimedia devices sensed by the interface controller, memory means storing a network monitoring table to be used in the signal processing means where it is looked up or updated, and a second system controller for communicating with the main function unit of the multimedia controller and for controlling the second interface control unit.

(0011)

(Function)

Such a control system for multimedia devices is able to detect any changes in the status of links between the multimedia controller and the multimedia devices without recourse to the user's constant nervous helps. Even if the event of adding a multimedia device or removing it frequently occurs, the productivity of multimedia over all the devices in the network can be improved.

(0012)

(Embodiments)

The present invention is next described in connection with embodiments of the system control method thereof by reference to the drawings.

(0013)

Let us first explain about the environment of object embedded multimedia devices connected to one another in a network.

(0014)

Fig. 1 shows a logic of connection of a multimedia controller with multimedia devices employing an object oriented concept of the invention. The multimedia controller 1 lies at the center. For the multimedia devices 2, respective individual communication paths to the multimedia controller 1 are established so that direct conversation of various items of information can be made between the multimedia controller and the multimedia devices in one to one basis. It will be appreciated that the control is made by transmitting messages to each other over that communication path. The term "multimedia devices" herein used means, specifically speaking, CD players, digital VTRs, digital cameras, digital TV sets and other AV devices, and digital FAX, digital copiers, printers and other OA devices, that is, all of those devices which deal with multimedia data.

(0015)

The controller is assumed here to be part

of hardware dedicated solely to this purpose. But it is also possible to realize an equivalent controller by installing an especial OS and a particular application on the commonly available processor in the personal computer or word processor.

(0016)

Referring next to Fig. 2, there are shown three configurations (a) to (c) for physically connecting a multimedia controller to a number of multimedia devices to establish the respective duplex communication paths.

(0017)

The daisy chain line of Fig. 2(a) is employed in SCSI bus (ANSI X3.131-1986). The star configuration of Fig. 2(b) is employed in Ethernet (IEEE 802.3) 10BaseT. The multipoint line of Fig. 2(c) is employed in Ethernet 10Base2/5.

(0018)

It is also to be noted that with regard to another possible configurations, there is GPIB (IEEE 4888) as obtained by mixing the (a) to (c). Even in Ethernet, the (b) and (c) may be mixed. With regard to another possible communication systems, there are optical fiber cables and ISDN. So it is to be understood that, besides those of Fig. 2, many other combinations are possible to make and may be selectively employed as desired.

(0019)

How to establish such duplex communication paths and which to select are not essential to the

invention, so no particular remarks are given except that, as the communication system differs from one to another, some physical limitations are laid on the transfer speed, the number of connected devices, the length of the cable, the shape of the connector, etc. For the each other's transmission of messages, because the protocols have their hierarchies differentiated from each other, these limitations are out of question. In order to insure that the peripheral devices each are connected to the controller reliably and accurately, however, there is need to provide for the system with at least one physically (mechanically and/or electrically) common interface.

(0020)

To realize high speed transmission of data such as those of motion pictures, it is recommended to employ what is faster than Ethernet, that is, the optical transmission such as FDDI (Fiber Distributed Data Interface) or B-ISDN. But, in here, for the purpose of simplicity of explanation, discussions are conducted on assumption that Ethernet 10Base2(/T) which, because of its cheap price, is widely used is adopted as the common communication connector.

(0021)

The internal pieces of hardware of the usual multimedia device are shown in an block diagram of Fig. 3.

(0022)

A plurality of multimedia devices are connected

via a LAN 4 to the controller. Now this LAN is Ethernet so each of these devices is provided with an interface 20 for implementing its protocol (TCP/IP). This can be realized by using an exclusive LSI or the like. It is in here that the transmitted message itself is taken out. Conversely it is from here that a message is sent out to the controller. An example of these messages, if in the Objective-C, is given, as the general format, by the following expression:

(0023)

(Destination Object Method Name: Parameter)

In another languages, the expression takes different styles, but is similar in the following basic components:

(0024)

(1) Addressing a terminal object;

(2) Selecting a method (instruction to execute); and

(3) Putting data in parameters, if any.

How to deal with this message is described in connection with the flow of software of Fig. 23.

(0025)

In the interior of the multimedia device, a CPU 11 processes all software and controls all hardware through an external bus 10. The programs, the initial values and proper data are stored in a ROM 12. To

temporarily store data and internal parameters such as those representing the device status, there is a RAM 13. When executing the programs, this RAM 13 is used as a work area. A data I/O 14 is used in accessing multimedia data stored on an internal or external medium 15. A mechanical system driving portion 16 controls mechanical parts 17 such as an electric motor. An electrical system driving portion 18 controls electrical circuits for switches SW and indicators such as LEDs. As the multimedia data are of digital form and range from pictures to sound to texts, it is possible for the medium 15 to take various types, namely, optical disks such as CD-ROM and MD, magnetic tapes such as DCC and DAT, and semiconductor memory cards.

(0026)

Referring next to Fig. 4, a block diagram shows the internal hardware aspect of the multimedia controller 1. Connections to the multimedia devices are established via the LAN 4. Now this LAN 4 is Ethernet, so there is an interface 31 for implementing its communication protocol (TCP/IP). This can be realized by using a LSI or the like solely dedicated thereto. It is in here that the transmitted message itself is taken out. Conversely it is from here that messages are sent out to the multimedia devices.

(0027)

In the interior of the multimedia controller 1, a CPU 21 processes all software and controls all hardware

through an external bus 30. The programs, the initial values and proper data are stored in a ROM 22. To temporarily store data and internal parameters such as those representing the device status, there is a RAM 23. When executing the programs, this RAM 23 is used as a work area. A multimedia filing device 25 performs storing, retrieving, reproducing or editing of multimedia data, regardless of whether filing device is an internal or external medium. Accessing to it is controlled by a data I/O 24. An electrical system driving portion 28 controls electric circuits for switches SW and LEDs or like indicators. A display 27 constitutes a man-machine interface. Its displaying operation is controlled by a controller 26. There is further included a mouse or like pointing device, though not shown.

(0028)

Fig. 6 is a diagram of the hierarchy of systems in the software aspect of the multimedia device. The internal block diagram of Fig. 3 refers to hardware 57. An OS 58 is fundamentally in charge of this hardware. What type of OS to select is not itself particularly limited, but it is desired that the real-time facility and the multitasking capabilities that run more than one program in parallel at a time are available in combination. On this OS, the multimedia device has a class library 59 which is differentiated from the other devices in order to realize embedding of an object into that multimedia device.

(0029)

Though not shown, the multimedia device has another library concerning the control panel for itself and the control program. When connected to the multimedia controller, this library is transferred thereto, thus permitting a specific control to the multimedia device to be made at the side of the controller. There is also a C function 60 to be used as a timer and in performing arithmetic computations.

(0030)

At the top of the hierarchy, there is an application software 61 which takes its part in controlling the main system of the multimedia device and communicating with the multimedia controller, and as the user interface. The use of this application makes it possible that the main system of the multimedia device as one object is controlled in a variety of ways by transferring messages from or to the controller, and that the internal parameters are read as instance variables and altered.

(0031)

Fig. 5 is a diagram of the hierarchy of systems in the software aspect of the multimedia controller. The internal block diagram of Fig. 4 refers to hardware 50. An OS 51 is fundamentally in charge of this hardware. In here, too, what type of OS to select is itself not particularly limited. But it is desired that the real-time facility and the multitasking capabilities

are available in combination.

(0032)

On this OS there is a window server 52 which is in charge of the whole of a GUI (Graphical User Interface) that displays a plurality of control panels for the connected multimedia devices and the status of all links in the network on the screen, and controls coordination of inputting and outputting of data. A common class library 53 stores what has been made ready beforehand in the controller by itself, that is, a set of basic and common components (in the form of objects) concerning the user interface, namely, buttons, slide volumes and text presentation areas and also concerning the control.

(0033)

For every multimedia device, on the other hand, a specific class library 55 stores a set of components (in the form of objects) concerning the unique panel display and control. This specific library, as described before, increases its content each time one more multimedia device is brought into connection with the system, as an additional set is sent from that device. This procedure will be described later in more detail. There is also a C function 54 for the timer and arithmetic computation. At the top of the hierarchy, there is an application software 56 which undertakes in controlling coordination of all the connected multimedia devices and communicating with these multimedia devices and that

functions as the user interface.

(0034)

The flow of control signals and the transmission of messages between this controller and the multimedia device are described below.

(0035)

Fig. 7 shows a state of the system before the multimedia device is connected to the multimedia controller. In Fig. 7, digital data are transmitted over a communication line or LAN 4. A multimedia controller 1 controls operations of all parts of the system. A multimedia device 2 to be connected to the LAN 4 is shown with its structure of construction in generalized form. 205 is the one of software objects (hereinafter abbreviated to "objects") which always resides in the multimedia controller 1 and coordinates all parts of the system, named "system director".

(0036)

1064 is an object which functions as a multimedia device and which is differentiated from the other objects on the LAN 4, named "multimedia device". This object is further comprised of three objects 1065, 1066 and 1067.

(0037)

The "controller for multimedia device" object 1065 is in charge of hardware to realize a majority of functions of the multimedia device 2. The "data input to multimedia device" object 1066 is used for entering

digital data as transmitted from the other devices over the LAN 4. The "data output from multimedia device" object 1067 is used for transmitting digital data to the other devices over the LAN 4.

(0038)

When the multimedia device 2 is connected to the multimedia controller 1 via the LAN 4, an object that stands in the place of that multimedia device 2 must be formed in the multimedia controller 1. To describe this "deputy multimedia device" object, a specification is written in a file 1061. This file comprises a section 1062 for the specification of a control panel for the multimedia device 2 and another section 1063 for the specification of an input or output of data to or from the multimedia device. In particular, the section 1062 for describing the "control panel for the multimedia device" object realizes the function of describing the control panel so that one interacts with the multimedia device 2 by means of a GUI, that is, the function of the language in which to describe the GUI.

(0039)

Fig. 8 is a diagram to explain the state of the system after the multimedia device 2 is connected to the LAN 4. In Fig. 8, what is now formed in the multimedia controller 1 is an object 1068. In the interior of the multimedia controller 1, this object acts as substitute for that multimedia device 2, so it is called "deputy" multimedia device. This object 1068 comprises

an object 1069 which functions as the control panel for the multimedia device 2, named "control panel for multimedia device", another object 1070 which, when to input data, functions as substitute for the "data input to multimedia device" object 1066, called "deputy" object, and still another object 1071 which functions also as substitute for the "data output from multimedia device" object 1067, called "deputy" object.

(0040)

Fig. 9 is a diagram of the structure of the common class library. For the objects having a similar feature, their common attribute and function have to be defined. To this purpose, a class, say a first class 1079, functions as a template. The first class 1079 to the p-th class 1085, totaling p classes, are summed up in a library 1086. This is conventionally called "class" library. All the objects belong to the respective specified classes. The type and name of data for the internal variables and the type and name of data for the internal functions representing the data processing means (usually called "class methods" the one of the objects which belongs to a particular one of the classes should possess are defined in a portion 1080. To allow one to access to the class methods, all the codes of the class methods are cited in the form of a table 1081 relative to a pointer. A first function code 1083 through a k-th function code 1084, totaling k function codes for the k class methods, are stored in a "code" portion 1082.

(0041)

Fig. 10 is a diagram of the structure of a typical object. In Fig. 10, an object 234 comprises a portion 244 for accommodating the pointer that is to go to the class method table, communicating means 245 for messages, processing and retrieving means 246, a portion 239 for methods and a portion 235 for internal data. The "method" portion 239 comprises beginning with first data processing means 240, second data processing means 241 and so on and terminating at m-th data processing means 242, totaling m data processing means. The portion 235 for internal data comprises beginning with a first internal data 236, a second internal data 237 and so on and terminating at an n-th internal data 238, totaling n internal data.

(0042)

All the internal data in the portion 235 differ with the different objects and, therefore, are left as they stand in the interior of each of the objects. The data processing means in the "method" portion 239, on the other hand, can be used in common by the other objects, if of the same class. To assure this, therefore, a class method table 243 is provided so that the first data processing means 240 through the m-th data processing means 242 are made manageable by each of the classes. Thus, such common data processing means are shared by a number of objects which belong to the same class. To look up the class method table 243 from every object,

the pointer is brought into this table from an accommodating portion 244 therefor.

(0043)

The message communicating means 245, when in receiving a message from another object, transfers it to the processing and retrieving means 246, where the message is analyzed to identify its address section and the corresponding one of the data processing means to it is retrieved from the "method" portion 239 (virtually the class method table 243). As the selected data processing means operates, the data section of the message, the internal data from the portion 235 and some external data are processed in a predetermined way. In some case, such processing will result in production of a message. If so, this message is transmitted from the communicating means 245 to that other object.

(0044)

Fig. 11 is a diagram of the structure of the system director object 205. A space 1072 accommodates a pointer to a class method table 1073. 1047 is an object forming means for producing the "deputy multimedia device" object 1068 based on the information from the file 1061. Coordination of inputting and outputting data between the objects is controlled an data I/O means 343. Another object forming means 380 produces various application objects of different aims. Further included are communicating means 1074 for messages, processing and retrieving means 342, a "method" portion 1075 and an

"internal data" portion 1076. The internal data are an object ID 1077, control data 344 to be used in establishing links between any two of the multimedia devices when to carry out certain operations, and object record data concerning the connected multimedia devices or the produced objects therefor.

(0045)

When a multimedia device 2 is connected to the LAN 4, using the "deputy multimedia device" object forming means 1047, the system director object 205 reads the file 1061 for describing that object. From the information obtained from this file 1061, it then determines which class the object that should be produced belong to. Based on the definition in the portion 1080 of the corresponding class in the class library 1081, the "deputy multimedia device" object 1068 is made up.

(0046)

Fig. 12 is a diagram of the structure of the "control panel" written section of the "deputy object" written file. In Fig. 12, the section 247 for describing a control panel object comprises first to i-th databases 248 to 249 to be used in describing i articles of the object. One database consists of data 250 for recognizing the object, data 254 for drawing the object and data 260 for object linkage.

(0047)

The data 250 for recognizing the object are of a name 251 of the class which the object belongs to,

a unique ID 252 to the i-th article, and an ID 235 of the object to which the i-th article is appendant.

(0048)

The data 254 for drawing the object are used to depict a button or like object constituting part of the control panel window 231 on the screen, comprising first to j-th packs 255 to 259 of data for drawing j articles of the object, One pack consists of data 256 for the location and size at and to which to depict the object, data 257 for the pattern and color and an object image 258.

(0049)

The data 261 for object linkage provide information about the links in the network and are used when one of the items constituting the control panel, for example, the controller object 207, is to connect to another, say relational object, comprising first to k-th data 261 to 264 for establishing k links between the items. One data are of an ID 262 of a relational object, and an message 263 for transmission to the relational object.

(0050)

Fig. 13 is a diagram of the structure of a "deputy data I/O object" section in the "deputy object" file. In Fig. 13, the "deputy data I/O object" section 650 includes first to m-th data 651 to 652 for producing m articles of a deputy input object. Each data are of an ID 652 of its own article, an ID 653 of a link

terminal "data input" object and a compatible file type list 654. The section 650 further includes first to n-th data 659 to 663 for n articles of a deputy "output" object. Each data are of an ID 660 of its own article, an ID 661 of a relational object, and a compatible file type list 662.

(0051)

Next, taking an example of a digital VTR to which the above-described system control method of the invention is applied, the operation of the control system for multimedia devices 2 is described below.

(0052)

Fig. 14 is a diagram showing the state of the control system before an object embedded digital VTR is connected to the multimedia controller. In Fig. 14, as an object is embedded into the digital VTR 203, this "digital VTR" object 206 always resides in the digital VTR 203 and functions as the object embedded digital VTR as viewed from the other devices on the LAN. The digital VTR object 206 is further constructed from three objects. Of these, a "digital VTR controller" object 207 controls the hardware of the digital VTR 203.

(0053)

Another or "data input to digital VTR" object 208 is used to input digital data as transmitted from the other devices over the LAN 4. The other or "data output from digital VTR" object 209 is used to output digital data for transmission to the other devices over

the LAN. When the digital VTR 203 is connected to the multimedia controller 1 via the LAN 4, a deputy digital VTR object is generated in the interior of the multimedia controller 1 based on the information from a "deputy digital VTR object" written file 210.

(0054)

This file 210 comprises a section 211 in which the specification of a control panel for the digital VTR 203 is written, or which is used in describing a "control panel for the digital VTR" object and another section 212 in which the specification of an object that acts as substitute for the data input/output of the VTR 203, or which is used in describing the "deputy data I/O of digital VTR" object.

(0055)

Fig. 15 is a diagram of the structure of the VTR controller object 207, where a portion 1009 accommodates a pointer that goes to a class method table 1018. This table is formed with a wide variety of data processing means including reproducing means 1019 for operating the play mode under the control of the hardware of the VTR and recording means 1020. 1010 is communicating means for messages. 1011 is processing and retrieving means. Though a "method" portion is shown at 1012, it is in actual practice that the data processing means are presented by a class method table 1018. The internal data in a portion 1015 are of many variables and status information necessary to control the digital

VTR, for example, the tape running speed 1016 and the current tape footage 1017.

(0056)

Let us first explain an operation which occurs when the digital VTR 203 is connected to the LAN 4. Fig. 16 is a flowchart of the routine for this operation. Fig. 17 is a plan view of a window for the multimedia controller 1 on the screen. In Fig. 17, the multimedia controller window 228 contains a number of icons of which the icon 229 appears when the digital VTR 203 is connected to the LAN 4. Using a mouse or like pointing device, one can choose a location with a cursor 230. The pointing device, though not shown, is provided with buttons. The user presses the button and then releases it. This pressing once is usually called clicking. Pressing it twice in quick succession is called double clicking. Incidentally, as other usable devices, mention may be made of a camera (for inputting still pictures), a tuner, a television set, various relational databases and a CD. To allow the user to select these options, the window 228 displays their icons.

(0057)

Fig. 18 is a diagram to explain the state of the system when the object embedded digital VTR 203 as an example of the multimedia device is connected to the LAN 4. In Fig. 18, an object 220 is now formed in the interior of the multimedia controller 1. This object serves as substitute for the digital VTR 203, being named

"deputy digital VTR". The deputy digital VTR object 220 is constructed from a deputy "control panel for digital VTR" object 221 which functions as a control panel for the digital VTR 203, another or deputy "data input to digital VTR" object 222 which, when inputting data, functions as substitute for the data input object 208, and yet another or deputy "data output from digital VTR" object 223 which functions also as substitute for the data output object 209.

(0058)

Referring now to Fig. 16 and Fig. 17, the routine for such an operation is described below. When the digital VTR 203 is connected to the LAN (636), the system director object 205 detects establishment of a new connection (637). Then it sends a device ID to the digital VTR 203 (638).

(0059)

Using the deputy multimedia device object forming means 1047, the system director object 205 then loads the "deputy digital VTR object" file 210 from the digital VTR 203 (639). Based on the information from the file 210, the system director object 205 then generates a deputy digital VTR object 220 in the interior of the multimedia controller (640) by using the deputy multimedia device object forming means 1047. Such a procedure results in a change of the status of the network as shown in Fig. 18. Then, the deputy digital VTR object 220 presents the display of an icon 229 for the

digital VTR 203 in the multimedia controller window 228 (641). After this, the system stands by for instructions from the user (642).

(0060)

Subsequently, with the help of the window displayed on the screen based on the "control panel for digital VTR" object of the multimedia controller, the user will activate the digital VTR. On this, the control system can operate the digital VTR through the intermediary of the deputy digital VTR object 220 in the multimedia controller 1.

(0061)

Next, the relationship between the content of the "deputy digital VTR object" written file 210 and the object to be generated is described in detail below.

(0062)

Fig. 19 shows an icon for the VTR 203 and Fig. 20 shows an example of the control panel window on the screen. This icon 229 of Fig. 19 appears when the digital VTR 203 is connected to the LAN 4. The digital VTR control panel object 221 depicts a window of Fig. 20 by default on the screen. In this window, there is an option menu 232 for selectively display the control panel windows on the screen. As the tape is running, the passed time is displayed in a time counter box 265. As the digital VTR 203 has a number of control modes, there is a mode option box 267 containing a first switch button for setting a control mode by default and a second

switch button 268 for selectively setting more elaborate control modes. A rewind button 269, a reverse play button 270, a pause button 271, a play button 272, a fast feed button 273, a stop button 274 and a record button 275 are displayed in array.

(0063)

Fig. 21 is a diagram, partly in pictorial form, to explain the correspondences between the classes the objects belong to and the constituent elements of the "control panel for digital VTR" object. All classes to which the fundamental constituent elements belong are defined previously in the class library 1081. This library is kept in the multimedia controller 1. As is obvious from Fig. 21, all the constituent elements of the "control panel for digital VTR" object 221 function as respective individual objects constituting the "control panel for digital VTR" object 221.

(0064)

In Fig. 21, the frame of a control panel window 231 on the screen corresponds to the VTR control panel object 221 (ID = 1) of panel class. The control panel display option menu 232 corresponds to the panel view setting menu object 285 (ID = 2) of menu class. The time counter box 265 corresponds to the time counter object 286 (ID = 3) of form class. The rewind button 269 corresponds to the rewind button object 287 (ID = 4) of button class. The reverse play button 270 corresponds to the reverse play button object 288 (ID = 5) of button

class. The pause button 271 corresponds to the pause button object 289 (ID = 6) of button class. The play button 272 corresponds to the play button object 290 (ID = 7) of button class. The fast feed button 273 corresponds to the fast feed button object 291 (ID = 8) of button class. The stop button 274 corresponds to the stop button object 292 (ID = 9) of button class. The record button 275 corresponds to the record button object 293 (ID = 10) of button class.

(0065)

The control mode dialog box 266 corresponds to the control mode selection object 294 (ID = 11) of button group class. The first switch button 267 corresponds to the default button object 295 (ID = 12) of radio button class. The second switch button 268 corresponds to the advanced button object 269 (ID = 13) of radio button class.

(0066)

Next, of the objects constituting the "control panel for digital VTR" object 221 shown in Fig. 21, an example of the play button is taken to explain how to make up an object 290 therefor.

(0067)

Fig. 22 is a diagram to explain the formation of the play button object 290. In Fig. 22, 297, 298, 299, 300, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610 and 611 show the elements written in the "control panel object" section of the "deputy digital VTR object"

file 210.

(0068)

A piece of information 297 for recognizing the object are formed with a class name 298, an object ID 299 and a superior object ID 300. A first piece of information 601 for drawing an article of the object are formed with data 602 for location and size, data 603 for pattern and color and an object image 604. A second piece of information 605 for drawing an article of the object are formed with data 606 for location and size, data 607 for pattern and color and an object image 608. A piece of information 609 for object linkage are formed with a link terminal object ID 610 and a message 611 to transmit.

(0069)

The play button object 290 of button class is generated by information from that class and the "'control panel of objects' object" section 247 in the "deputy digital VTR object" file 210. A portion 613 accommodates a pointer that goes to a class method table 625, where it points to methods in the button class. The button class method table 625 is formed from means 626 responsive to start of generation of an object of button class for initializing the internal variables of the button object are initialized, means 627 for depicting the button object to display, and click response means 628. The user moves the mouse or like pointing device to position the cursor 230 on top of the play button.

Responsive to clicking on this button, the click response means 628 changes the display of the button for a moment to inform the user of the fact that the button object has been activated and sends a message to another object.

(0070)

The definition of every data processing means in the method table for these button classes is described in each class. Therefore, not only the play button object 290 but also all the other objects which belong to the button class share the common button class method table. 614 is message communicating means; 615 is processing and retrieving means; 616 is a portion for methods; and 620 is a portion for internal data. The internal data are of an object ID 621, the state 622 of the button, drawing parameters 623 and link data 624. The one of the types of the internal data which not only the play button object 290 but also all the other button objects which belong to the button class should possess is described in the class.

(0071)

The system director object 205 first reads in the " deputy digital VTR object" file 210, when to make up any of the objects. In the example of Fig. 22, it then accesses the data for recognizing the object and, on the basis of the description of its class name 298, forms the objects of the button class. For the play button object 290 to generate, the system director object 205 cooperates with the button initializing means 626 to

initialize the "internal data" portion 620. According to the example of Fig. 22, the object ID is set to ID = 7 by the description of the object ID 299. From the description of the superior object ID 300, the system director object 205 recognizes that the play button object 290 belongs to the "control panel for digital VTR" object 221. In such a manner, based on the information from the object to any of the objects of principal interest, the system director object 205 recognizes which one of the objects contains the other. Thus, a number of the constituent objects are put together to form a complex object.

(0072)

The button drawing means 627 depicts the play button object 290 on the basis of the drawing parameters 623 and the data 622 of the state of the button. The button drawing means 627 is automatically activated when the button object is generated and when the superior object moves.

(0073)

The first piece of information 601 describes that article 625 of the button which is effective in the situation when it is not pushed. The data 602 for location and size are used to determine a location at which the play button object 290 is depicted in the window for the "control panel for digital VTR" object 221 and a size to which a rectangular frame of the play button 625 is limited. To figure the rectangular frame, as the

button is not pressed, the data for drawing the button article 625 are expressed in the coordinates of the "control panel for digital VTR" object 221. Suppose, for example, the left hand upper and right hand lower corners are taken into account, then the data of that rectangular frame have a form like (X1, Y1) and (X2, Y2). For the play button object, when not pressed, its pattern and color are determined based on information either from the pattern and color data 603 or from the object image 604. The data 603 for pattern and color are described in an appropriate language to draw lines and paint colors, that is, to describe the object in a graphical form. The object image 604 is expressed in the form of bit map data. In general, the expression by the former costs a less amount of data, but the latter has rather a high degree of freedom.

(0074)

The second piece of information 605, similarly to the first piece of information 601, describes that article 626 of the button which is effective in the situation when it is pressed. Based on both of the first data 601 and the second data 605, the values of the drawing parameters 623 are determined. The link data 624 are set in based on information from the data 609 for object linkage. Therefore, as the message to transmit, "play" is set in and, as the link terminal object ID, a link terminal object ID is set in. In connection with the latter, it is to be noted that when to transmit the

message, only one of the terminal objects over the entirety of the system should be selected to receive this message. To this purpose, the link terminal object ID to be used is set in the preceded form by the device ID the system director object 205 has assigned to the digital VTR, when the digital VTR 203 was connected to the LAN 4.

(0075)

Even if it happens that two devices have their link terminal objects to use the same ID, therefore, the message can be transmitted right to the desired object. The button state data holds the information of whether or not the button is pressed.

(0076)

Fig. 23 is two flowcharts, one of which shows the operation when the user positions the cursor 229 on top of the icon 229 of the digital VTR 203 and double clicks, and another one which shows the operation when the user has manipulated the control panel.

(0077)

Fig. 24 is a plan view of the display of a window for the multimedia controller 1 on the screen as presented when the user has double clicked on the icon 229 of the digital VTR. In Fig. 24, the control panel window 231 for the VTR 203 is selected by default, and the play button is shown at 272.

(0078)

Fig. 29 is a diagram of the relationship between

the structure of the "control panel for digital VTR" object of panel class and the object description data.

(0079)

In Fig. 29, a portion 1401 accommodates a pointer that goes to the class method table, in this instance, a panel class method table 1402. This table is formed with panel initializing means 1403 for initializing the panel object, panel drawing means 1404 for showing the panel in a graphical form, and click response means 1405 for activating the clicked object. Message communicating means 1406, processing and retrieving means 1407, and a portion 1410 for internal data are shown. The internal data are of an object ID 1411, the panel state 1412, and drawing parameters 1413. The "internal data" portion 1410 is initialized according to the description of the "deputy digital VTR object" file 210. The "'control panel for digital VTR' object" section 211 in this file 210 comprises data 1414 for recognizing the object, a first pack 1418 of data for drawing an icon 1426 of the digital VTR 203, and a second pack 1422 of data for drawing the frame of the control panel window for the digital VTR. The data 1414 for recognizing the object are of a class name 1415 (panel class), an object ID 1416 (ID = 1), and a superior object ID 2417. The first data pack 1418 consists of data 1419 for location and size, data 1420 for pattern and color, and an object image 1421. The second data pack 1422 consists of data 1423 for location

and size, data 1424 for pattern and color and an object image 1425.

(0080)

Referring to Fig. 23 and Fig. 29, the routines for displaying the control panel window for the digital VTR 203 and activating the play mode are described below. As described in connection with the routine of Fig. 16, at a time when the system director object 205 has generated the deputy digital VTR object 220, the deputy digital VTR object 220 presents the display of the icon 229 as obtained based on the icon image 1426. For now, when the user double clicks on the icon 229 for the digital VTR (643), the control panel object 221 of the deputy digital VTR object 22 sends a message of executing the drawing function to all objects constituting the control panel object 221. Based on this message, all the objects shown in Fig. 21 activate the drawing means. Meanwhile, the control panel object 221 depicts the frame of the control panel window for the digital VTR based on the second pack of object drawing data. As a result, the digital VTR control panel window 231 is displayed on the screen (644) as shown in Fig. 24 and waits for instructions from the user (645). With this, when the user positions the cursor 230 on top of the play button 272 and clicks (646), the control panel object 221 sends a message "PLAY" to the controller object 214 of the digital VTR 203 (647). Responsive to this message, the controller object 214 of the digital VTR 203 activates

the play executing means (648), thus starting an operation of the play mode of the digital VTR 203.

(0081)

As has been described above, according to the invention, when a multimedia device is only connected to the multimedia controller via the LAN, its object necessary to coordinate the multimedia device with the others, or deputy multimedia device object, is automatically generated in the multimedia controller. Further, the control panel necessary to choose the multimedia devices is automatically displayed in the multimedia controller window on the screen. With the help of this control panel, the user activates an item. Then an unique message is transmitted to the controller object of the corresponding multimedia device. So the desired functions are executed. Since the information necessary to generate the deputy multimedia object, which in turn is necessary to manipulate the multimedia device, is obtained from the "deputy multimedia device object" written file as read from the multimedia device, what suffices for the multimedia controller is only the fundamental class library. So there is no need to store the related database to any specific multimedia device in advance.

(0082)

Fig. 25 is a diagram of the relationship between the structure of the deputy "data input to digital VTR" object and the data for describing the object. In

Fig. 25, the deputy "data input to digital VTR" object 222 contains a portion 668 for accommodating a pointer that goes to the class method table, in this instance, a deputy data input class method table 679. This table is formed with means 680 for initializing the deputy data input object, means 681 for updating the link data and compatible file type reply means 678.

(0083)

669 is message communicating means; 670 is processing and retrieving means; 671 is a portion for methods; and 674 is a portion for internal data. The internal data are of an object ID 675, another ID 676 which represents the related data input object, compatible file types 677 and links 1006 with data output objects.

(0084)

The "deputy digital VTR object" file 210 contains a "deputy 'data input/output of digital VTR' object" section 212. Based on the information from this section, the deputy "data input to digital VTR" object is generated. The data for the "deputy input" object described in the section 212 are of an object ID 683 (in this instance, ID = 1), a related data input object ID (in this instance ID = 1), and a list 685 of compatible file types (in this instance, assumed to be formats so called "AV1" and "AV2"). According to the description of these parameters, the deputy input object initializing means 680 initializes the "internal data" portion 674.

(0085)

Fig. 26 is a diagram of the relationship between the structure of the deputy "data output from digital VTR" object and the data for describing the object. In Fig. 26, the deputy "data output from digital VTR" object 223 contains a portion 690 for accommodating a pointer that goes to the class method table, in this instance, a deputy data output class method table 1048. This table is formed with means 694 for initializing the deputy data output object, means 695 for updating the link data and compatible file type reply means 700.

(0086)

691 is message communicating means; 692 is processing and retrieving means; 693 is a portion for methods; and 696 is a portion for internal data. The internal data are of an object ID 697, another ID 698 which represents the related data output object, compatible file types 699 and links 688 with the data output object.

(0087)

Based on the information from the "deputy digital VTR object" file 210 at the "deputy 'data output from digital VTR' object" section, the deputy "data output from digital VTR" object is generated. For now, the data 1001 for the deputy data output object described in the section 212 are of an object ID 1002 (in this instance, ID = 1), a related data output object ID 1003 (in this instance ID = 1) and a list 1004 of compatible file types (in this instance, assumed to be formats so

called "AV1" and "AV2"). According to the description of these parameters, the deputy data output object initializing means 694 initializes the "internal data" portion 696.

(0088)

Fig. 27 is a diagram of the structure of the "data input to digital VTR" object. This object 208 includes a portion 1030 for accommodating a pointer that goes to the class method table, in this instance, a data input class method table 1031. This table 1031 is formed with file writing means 1032, data receiving means 1033 and link data updating means 686. 1023 is message communicating means; 1024 is processing and retrieving means; 1025 is a portion for methods; 1028 is a portion for internal data; 1029 is an object ID; and 1030 is link data.

(0089)

Fig. 28 is a diagram of the structure of the "data output from digital VTR" object. This object 209 contains a portion 1035 for accommodating a pointer that goes to the class method table, in this instance, a data output class method table 1044. This table is formed with file reading means 1045, data transmitting means 1046, and link data updating means 687. 1036 is message communicating means; 1037 is processing and retrieving means; 1038 is a portion for methods; 1041 is a portion for internal data; 1042 is an object ID; and 1043 is link data.

(0090)

After the deputy data input object 222 and deputy data output object 223 of the digital VTR 203 have been generated in the multimedia controller, these objects function as if they were chief ones, or the "data input to digital VTR" object 208 and the "data output from digital VTR" object 209. Now suppose the digital VTR receives, for example, a file from another multimedia device by the copy function, then the system director object 205 inquires of the deputy "data input to digital VTR" object 222 what types of files are possible to input. Responsive to this inquiry, the compatible file type reply means of the deputy "data input to digital VTR" object 222 gives off information about the file types the digital VTR 203 can accept.

(0091)

If the type of the file to be copied is found to be present among them, a link is established from the deputy output object of that multimedia device which has the file to be copied to the deputy "data input to digital VTR" object 222. The link data updating means 681 of this object 222 sends a message to the "data input to digital VTR" object 208. As the link updating means 686 of this object 208 is activated, the link data of the "data input to digital VTR" object 208 are updated.

(0092)

At the same time, the deputy data output object of that multimedia device which has the file to be copied

sends a message for updating the link data of the data output object. As the link data are updated, a link is established from the data output object of the multimedia device which has the file to be copied to the "data input to digital VTR" object 208.

(0093)

After this, the data transmitting means of the data output object of the multimedia device which has the file to be copied is activated. The data output object of the multimedia device which has the file to be copied sends a message to the "data input to digital VTR" object. As the data receiving means 1033 and the file writing means 1032 are activated, copying of the file is carried out. In short, when the user gives the copy command or any of the other commands to the deputy data input object and the deputy data output object in the multimedia controller, the deputy data input object and the deputy data output object send the messages to the data input object and the data output object of the main systems of the respective multimedia devices. Thus a link for data communication is established between these two multimedia devices. Concerning the operation of copying data, for example, it is not virtually necessary for the multimedia controller to take direct participation.

(0094)

As is understandable from the foregoing, according to the prior art, when to make control of the

entirety of a system having a plurality of multimedia devices connected to one another, the device drivers or like applications for this purpose had to be previously installed in the controller. According to the invention, however, such a necessity is obviated. So, when a multimedia device is only connected to the LAN, the control panel and the status of the network are automatically displayed in the controller window on the screen. With the help of the windows on the screen, therefore, it becomes easy to turn on and off the power sources of the devices, control the main systems of the devices, and coordinate inputting and outputting of the various signals and data.

(0095)

Another advantage is that of the items the multimedia devices have sent to the control panel of the controller, the ones which are identical in definition to the items the controller has already possessed may be exchanged either in part or all therebetween, depending on the user's taste. As the user interface differs with different makers, it is thus made possible to unitarily rearrange the items.

Yet another advantage is that it becomes possible to execute the control functions from a controller in the distant place and the functions of accessing to the terminal multimedia devices in a transparent fashion over the LAN.

(0096)

Next, using the above-described multimedia control system, an actual practice is made by connecting two or more multimedia devices to a multimedia controller and by controlling this network. In this case, a number of forms can be considered. Fig. 30 is a block diagram of first embodiment employing one of these forms.

(0097)

As is obvious from the same figure, the first embodiment of the invention is applied to a system comprising multimedia devices 102-1, 102-2 and 102-3 and a multimedia controller 1 for keeping track of all current connections of these multimedia devices.

(0098)

Though, in Fig. 30, the multimedia devices are shown three in number at 102-1, 102-2 and 102-3, any number of multimedia devices except one may be used of course. But, notice that there is necessarily an optimum largest number of devices to be used, as determined from the possible maximum load on the entire system. The aforesaid multimedia controller 1 is constructed with inclusion of a unit 103 for realizing the functions of the multimedia controller, an interface controller 107-0 for controlling communication physically or in low logic level, a main power circuit 108-0, a subsidiary power circuit 109-0, a timer circuit 105, a memory 104 storing a table for use in checking the status of individual links with the multimedia devices, and a system controller 106-0 for controlling these. The main power circuit 108-0

supplies electric power to the interface controller 107-0, the timer circuit 105, the system controller 106-0 and the memory 104. The electric power may be supplied, in some cases, from a dry battery. Otherwise, the communication line may be supplemented even with a power line, from which the electric power is supplied. The subsidiary power circuit 109-0 is arranged to turn on or off either when the user moves the main switch of the multimedia controller 1, or in response to request from the system controller 106-0. When turned on, electric power is supplied to the function unit 103 of the multimedia controller 1. This subsidiary power circuit 109-0 has its power source in the commercial power line at, for example, a socket for home use. The before-mentioned main power circuit 108-0 has, on the other hand, its power source in a separate system therefrom.

(0099)

With the use of two separate power supply systems as such at a time, it becomes possible that the mechanism driving system, because of its usually consuming huge electrical energy, is activated only when necessary. When not necessary, electric power little enough only to make communication can be supplied.

(0100)

The multimedia devices A, B and C each are constructed with inclusion of a function unit 111-1, 111-2 or 111-3, an interface controller 107-1, 107-2 or 107-3,

a main power circuit 108-1, 108-2 or 108-3, a subsidiary power circuit 109-1, 109-2 or 109-3, and a system controller 106-1, 106-2 or 106-3. The interface controllers 107-1, 107-2 and 107-3, the main power circuits 108-1, 108-2 and 108-3, the subsidiary power circuits 109-1, 109-2 and 103-3, and the system controllers 106-1, 106-2 and 106-3 are equivalent to the respective parts 107-0, 108-0, 109-0 and 106-0 of the multimedia controller 1.

(0101)

As a specific example of one of the multimedia devices A, B and C, let us take a digital VTR as shown in a block diagram of Fig. 31. The digital VTR includes a function unit 111 which is representative of the multimedia device function units 111-1, 111-2 and 111-3. This unit comprises, an analog output terminal 113, an analog input terminal 114, a D/A conversion circuit 115, an A/D conversion circuit 116, a frame memory 117, a signal processing circuit 118, an encoding circuit 119, a decoding circuit 120, a tape deck unit 122, a tape interface controller 121, a high speed data bus 123 and a control bus 124.

(0102)

Next, the operation of the first embodiment of the invention is described. Fig. 32 is a flowchart of the main event loop for the multimedia controller 1. The flow begins with a step (s101) of testing if an event occurs as the user who is managing the network has

actuated the multimedia controller 1, or any one of the multimedia devices has transmitted a control signal over the communication line. If so, the program for this event is then executed (s107). If there is no event, then examine whether or not a counter, g, has counted down to Check Time = 0 (s102).

(0103)

This counter, g, makes determination of a period in which the multimedia controller 1 recycles the operation of checking the status of the network. The timer circuit 105 described before is used to drive it. If the examination (s102) results in g Check Time = 0 then start checking any changes in the status of the network (s103). If no changes are found, device IDs are assigned to the multimedia devices which have been recorded in the status monitoring table (s104). The term "device ID" used herein means, speaking in terms of the TCP/IP protocol, what is equivalent to IP address. That is, it is a so-called logic address. By doing such assignment, without having to be conscious about the physical addresses, the multimedia controller can communicate with each of the multimedia devices.

(0104)

When a new device is connected to the network, or when any one of the current connections is removed therefrom, the aforesaid assignment of object IDs is automatically revised. Thus, selection of all the addresses to be used in transmission can be unitarily

controlled. Looking up the thus-obtained table, rewrite the display of the status of links (s105). A value of g Check Time is then set to the timer (s106). This value is named "PreSet". The network status monitoring table described above has a structure shown in Fig. 33.

(0105)

In Fig. 33, a column labeled "Unique Number" contains numbers dedicated solely to the respective individual multimedia devices. These numbers also serve as the physical addresses. This unique number is assumed also to be associated with information representing the attributes of the device, that is, categorizing the device to, for example, digital VTR or CD player. Suppose that another device of the same attribute is connected to the network, then record this connection by using a different name in an entry of a device name column in the table of Fig. 33. For example, the case of Fig. 33 has two digital VTRs connected. So, they are differentiated from each other as "Digital VTR1" and "Digital VTR2". The last column labeled "Address of location for Object Data" is used to record data for displaying all the multimedia devices and to accommodate a pointer to the area in which the control objects are stored. Concerning this, detailed explanation will be given later. The above-described device IDs are recorded in a column labeled "Device ID" in Fig. 33. a "Power On/Off" column and a "Link Check" column, too, will be explained later.

(0106)

Next, the process for checking changes in the status of the network in the above-described step (s103) is explained in great detail. Fig. 34 is a flowchart of the routine for this checking operation. To begin with, the variable named "Wait Timer" for counting the time-out period of communication is initialized (s107). The multimedia controller 1 then sends a message for detecting any change in the status of link to every one of the multimedia devices (s108). After the Wait Timer has been incremented (s109), whether or not the resultant value of the Wait Timer reaches the limit of the time-out period is examined (s110). If below the limit, then check to see if a reply message has arrived from any one of the multimedia devices (s111). If not, then return to the s109. If so, then inspect the network status monitoring table to find out whether or not that multimedia device which has transmitted this replay message was already recorded therein (s112). If so, then check the corresponding entry in the link check column of the table (s115). Otherwise, or if that device is found to be not yet recorded in the table, then record it in the table by checking the corresponding entry to this device in the link check column (s114).

(0107)

After that, initialize the Wait Timer (s116), causing a return to the loop start S109 and waiting for an acknowledgment from another one of the remaining multimedia devices. If it is determined in the aforesaid

step s110 that the counted time has reached the limit of the time-out period, then trace the network status monitoring table to see if, of the multimedia devices recorded therein, there is any one which is not remarked in the corresponding entry in the link check column (s117). If not, then finish the processing.

(0108)

If the s117 determines that there is the device which is not remarked in the corresponding entry in the link check column, then initialize the Wait Timer (s118) and resend the message for requesting the status of link to that device which is not remarked in the corresponding entry in the link check column (s119). After the Wait Timer has been incremented (s120), whether or not the resulting content of the Wait timer reaches the limit of the time-out period is examined (s121).

(0109)

If the s121 determines that the time-out period has expired, as this means that the multimedia device in question is no longer connected, then delete its entry from the network status monitoring table (s125). If it the s121 determines that the counted value of the Wait Timer is below the limit of the time-out period, then check to see if a replay message has arrived from that multimedia device (s122). If not, then return to the s120 and wait for the replay message from the multimedia device in question until the counted value of the Wait Timer reaches the limit of the time-out period sensed

in the step s121.

(0110)

If it is determined in the step s122 that the reply message has been received, then examine whether or not that reply coincides with what was expected at any rate of the device in question (s123). If so, then mark the corresponding entry in the link check column of the table (s124). If not, as this implies that something abnormal has happened, then delete the corresponding entry to that multimedia device out of the table and give a warning signal (s125). The flow then returns to the step s117. Such a procedure is repeated until there is no more entry left unchecked in the table. This sequence of the steps (s107 to s125) is carried out regularly. The entries in the network status monitoring table are thus updated to keep track of all current multimedia devices in the network.

(0111)

The multimedia devices to which the links are detected as normal are all presented to the user of the multimedia controller by the display means shown in Fig. 35. The user will give a command of displaying what devices are connected in the network. On execution of this command, the multimedia controller 1 displays an device collection panel window 126 on the screen of its own display device 125 as shown in Fig. 35A. In this figure, 127 is some application panel window which had been appearing before the user activated the icon for

the "display of available devices" command.

(0112)

On the device collection panel 126 there are displayed the icons (128, 129, 130, 131, 132, 133) for all the multimedia devices which have been detected as current connections. For the multimedia device whose subsidiary power source is turned on, its icon takes a form with the color inverted, or with the pattern framed. So the user can clearly see which of the devices are now powered on. In the example of Fig. 35(A), the icon 130 is shown as indicating that the subsidiary power source of the corresponding device is in OFF state.

(0113)

The user moves a mouse or like pointing device not shown to position the cursor 134 on top of one of the icons for the multimedia devices. On clicking, a control panel for the selected multimedia device by that icon is displayed in a window 135 on the screen as shown in Fig. 35(B). Hence this multimedia device can be manipulated as the user selectively actuates the switches on the control panel 135 by using the cursor. In the example of Fig. 35(B), the "power on/off" button is clicked to turn on the subsidiary power source. This is reflected to the item selection panel 126 as shown in Fig. 35(C).

(0114)

Fig. 36 to Fig. 40 are flowcharts of the aforesaid on/off operation of the power source. Fig. 36

is a routine for the multimedia controller, while Fig. 37 is a routine for each of the multimedia devices. When the user pushes the on/off button, the multimedia controller first sends a power supply message to the selected one of the devices (s126) and then resets a timer named "command wait counter" for counting the time-out period (s127) and waits for an acknowledgment. On receipt of the power-on request message, the terminal device executes the power supply function (s134), then test if the subsidiary power source has started operating normally (s135) and, as it is operating normally, then transmits a message informing of the normal completion of actuation to the multimedia controller (s136). If the actuation has failed due to a malfunction, then transmit an error message (s137). The multimedia controller 1 receives the reply message, so it is in the step s128 that it determines when the acknowledgment is received. Whether or not the start of the operation has completed normally is then examined (s131). If so, then turn on the corresponding entry in the "power on/off" column of the network status monitoring table (s133). If a malfunction has occurred, let the user know that an error has happened (s132).

(0115)

It is also possible to automatically start the on/off operation of the power source. The process for this is shown in the flowcharts of Figs. 38, 39 and 40. Fig. 38 is a routine for the multimedia

controller. Fig. 39 is a routine for the power-on operation in a selected one of the multimedia devices. Fig. 40 is a routine for the automatic power-off operation in each of the multimedia devices by itself.

(0116)

The multimedia controller first selects one of the multimedia devices and transmits a message for demanding start of the operation to it (s190). On receipt of this, the selected multimedia device tests if its own subsidiary power source is already turned on at that moment (s199). If in off state, then execute the "power on" function (s500) and test if the power source has started operating normally (s501).

(0117)

If the power source is turned on normally, or if the s199 determines that the power source has already been turned on, then carry out the operation that has been demanded from the multimedia controller (s502) and test if the start has completed normally (s503). If so, a message of the completion of normal start is transmitted to the multimedia controller. If an abnormality has happened, or if it is determined in the s501 that the power source has failed to start normally, then turn off the subsidiary power source (s505) and transmit a message of abnormally interrupting the start to the multimedia controller (s506).

(0118)

In the meantime, the multimedia controller

in itself, after having sent the message to the selected multimedia device in the aforesaid step s190, starts waiting for a reply to it. At first, it clears the command wait counter (s191) to start counting of a time-out period for the reply, and examines whether or not the reply has arrived (s192). If the reply is not received, then examine whether or not the content of the command wait counter has reached the limit of the time-out period (s193). If the limit of the time-out period is not exceeded, then increment the command wait counter (s194), causing a return to the s192, where to wait for the reply.

(0119)

If the s192 determines that the reply message is received, then test if this message is a message of the completion of normal start (s195). If normal, then check the corresponding entry to the selected multimedia device in the "power on/off" column of the network status monitoring table (s197).

(0120)

In the cases that, as a result of the test in the step s195, the received message is an abnormal start interrupt message and that, as a result of the examination in the aforesaid step s193, it is determined that the command wait counter has exceeded the limit of the time-out period, then informing the user of this by an error message (s197) and record "off" mark in the corresponding entry to the selected multimedia device

in the "power on/off" column of the network status monitoring table (s198).

(0121)

The automatic power-off operation: Each of the multimedia devices first examine whether or not there is any event left unprocessed (s507). If there is, then practice that event (s513). If no more event to process is left, then set a power-down counter for a preliminary time to the start of an automatic power-off operation (s514), and set a flag to true (s515). That counter is arranged to be decremented by a software or hardware timer. This flag indicates whether or not the power down counter is effective, being named "power down counter enable." Only when this flag is set in true, the automatic power-off operation is started by reference to the power down counter.

(0122)

If, as a result of the examination in the s507, there is no event, then examine whether or not the "power down counter enable" flag is set in true (s508). If so, then access the power down counter (s509). If its value is 0, then execute the function of turning off the aforesaid subsidiary power source (s510) and send a message of having turned off the power source to the multimedia controller (s511). After this, the power down counter enable flag is changed to false (s512) and the next event is waited for.

In the case that, as a result of the examination

in either of the s508 and s509, the power down counter enable flag was false, or the power down counter was not 0, no further processing is done and the next event is waited for. The multimedia controller, on the other hand, receives the aforesaid "power-off" message and records an "off" mark in the corresponding entry to this multimedia device in the power on/off column of the network status monitoring table.

(0123)

By making such provision for automatically controlling the on/off operation of the subsidiary power source of each of the multimedia devices, a necessary one of them only can be activated, and the power supply can be maintained only for a necessary time, thereby giving an advantage that the consumption of electric power is reduced to a minimum.

(0124)

It is also to be noted that instead of leaving it to the user's choice to request the device collection window 126 as shown in Fig. 35(A), all the icons for the current multimedia devices may be always displayed in a lower margin of the screen 125 as shown in Fig. 35(D). With this, when some of the current links are broken, the icons for the removed ones of the devices disappear. If this idea is employed, all current multimedia device can be displayed in real time fashion.

(0125)

(Another Embodiments)

Fig. 41 is a block diagram of the construction of a second embodiment of the invention. As will be seen from Fig. 41, the second embodiment of the invention is fundamentally similar in construction to the first embodiment, but one of its characteristic features is that the multimedia devices each are provided with a timer circuit 1 (136-1, 136-2 or 136-3). Again, the second embodiment is the same as the first embodiment in a point that it is regularly that any changes in the status of links are checked, but a difference lies in a point that, while the first embodiment uses the timer circuit in the multimedia controller for determining the recycling period, the second embodiment makes use of the timer circuits 1 (136-1, 136-2 and 136-3) which take their places in the respective multimedia devices.

(0126)

Another feature is that a network status monitoring table to be used in the second embodiment differs from that used in the first embodiment in a point that a column for link continuance detecting counters is added to the latter. This counter is provided by every one of the multimedia devices and arranged to be driven by the second timer circuit 2 in the multimedia controller. In the second embodiment of the invention, each multimedia device regularly transmits a message informing of continuance of its connection to the multimedia controller, thus manifesting the presence of itself. To this purpose, each time it has received the

connection continuance message, the multimedia controller resets the link continuance detecting counter in the corresponding entry.

(0127)

Since the link continuance detecting counters are incessantly driven to count by the timer circuit 2, if it has happened that any of the multimedia devices does not transmit the connection continuance message, the link continuance detecting counter will go on being incremented. At a time when the content of the counter has reached a threshold, the decision is made that this device is no longer connected, or has fallen in an abnormal condition.

(0128)

Fig. 43 is a flowchart of the event loop for the multimedia controller 1 and Fig. 44 is a flowchart of the event loop for the multimedia device. Using Fig. 43 and Fig. 44, the operation of updating the entries in the network status monitoring table in the second embodiment of the invention is described below.

(0129)

Every multimedia device first examines whether or not there are any events as the user has commanded with help of the control panel in the respective individual multimedia device or as are transmitted from the multimedia controller over the communication path (s148). If there is an event, then execute the corresponding function to that event (s149). If there

is no more event, then test if an "f" refresh counter = 0 is established (s150). This refresh counter regulates the cycle of transmission of a message informing of the current connection from the multimedia device to the multimedia controller, as it is driven to count by the respective individual timer circuit 136-1, 136-2 or 136-3. If it is determined in the s150 that the "f" refresh counter = 0 is established, then transmit the message informing of the current connection to the multimedia controller (s151) and reset the "f" refresh counter, causing a return to the s148 and waiting for the next event.

(0130)

Meanwhile, the multimedia controller examines whether or not there are any events as the user has commanded with help of the control panel thereof, or as are transmitted from any of the multimedia devices over the communication path (s138). If there is an event, then check to see if this event is the aforesaid message informing of the current connection transmitted from any of the multimedia devices (s141). Otherwise, then execute the corresponding function to that event (s143). If so, then inspect the network status monitoring table to determine whether or not that multimedia device which has given off this message is already recorded (s142). If not yet, then record that multimedia device in the network status monitoring table by adding a new entry therefor (s145) and clear a link continuance detecting

counter, or "g" refresh counter_X, in the added entry corresponding to that new multimedia device to zero. If the s142 determines that the multimedia device in question is the one already recorded in the table, then clear the "g" refresh counter_X in the corresponding entry to this multimedia device to 0.

(0131)

Based on the thus updated entries in the network status monitoring table, the current links in the network are displayed on the screen (s147). After return to the s138, the multimedia controller makes itself ready for the next event. If the s138 determines that there is no more event, then check all the link continuance detecting counters or "g" refresh counter_X's, from one to another (s139). If one of the multimedia device is found to have its "g" refresh counter_X reaching the limit, then delete the corresponding entry to this multimedia device out of the network status monitoring table (s140) and display all the current links in the network based on the updated entries in the aforesaid table (s147).

(0132)

As to the operation that follows the actuation of the on/off button for the power source and as to the forms of the icons for those multimedia devices which have been found to continue existing, similar means and effects to those of the first embodiment can be realized.

(0133)

A third embodiment of the invention is next described.

Fig. 45 is a block diagram of the construction of the third embodiment of the invention. As will be seen from Fig. 45, the third embodiment of the invention is fundamentally similar in construction to the first embodiment, but one of its characteristic features is that the timer circuit 105 in the multimedia controller 1 is no longer necessary, and use is made of a repeater 137 as arranged on the physical connection between the multimedia controller and a group of multimedia devices. The repeater 137 includes at least one network node terminal, node detecting circuits (138-1, 138-2 and 138-3) in one to one correspondence to the node terminals, an interface controller 107-4 which is similar to that in the multimedia controller or each of the multimedia devices, a power circuit 108-4 and a control unit 139 for coordinating those parts. The aforesaid detecting circuit observes the network status, for example, the traffic rate of communication to determine whether or not what is connected to that node is a communicable device.

(0134)

The third embodiment of the invention, unlike the first and second embodiments, does not set the cycle to a certain constant in which to regularly check the status of individual links in the network. Its characteristic feature is that, when a new multimedia

device is connected to the network, or when one of the current multimedia devices is cut off from the network, a message of this situation is transmitted from the one of the repeaters which is used to add or remove that multimedia device to the multimedia controller. In such a manner, the multimedia controller keeps track of all current connections in the network. It should be noted that a network status monitoring table to be used in the third embodiment is similar to that described in connection with the first embodiment of the invention by reference to Fig. 33.

(0135)

The operation of the third embodiment is described below. Fig. 47 is a flowchart of the main routine for the event loop of the multimedia controller 1 and Fig. 48 is a flowchart of the routine for the repeater to send a message informing of the establishment of a new connection to the multimedia controller. Fig. 49 is a flowchart of the routine for the repeater to send another message informing of missing one of the current connections to the multimedia controller.

(0136)

To begin with, the operation of the repeater is described using the flowchart of Fig. 48. The repeater examines the outputs of the aforesaid individual node detecting circuits in one to one correspondence to its own ports (s153) and determines whether or not a multimedia device has been added to any port (s154).

If the establishment of a new connection is detected, then send a message for informing this situation to the multimedia controller (s155) and clear an acknowledgment counter to wait for an acknowledgment from the multimedia controller (s156). Whether or not the acknowledgment has been sent from the multimedia controller is examined (s157). If no acknowledgment has been received yet, then increment the acknowledgment counter (s158) and test if the limit of the time-out period is reached (s159). If the time-out period does not expire yet, return to the s157 occurs, waiting for the acknowledgment from the multimedia controller. If the s159 determines that the time-out period has expired, then inform the user of the fact that an error has occurred (s160). If the s157 determines that the acknowledgment has arrived as normal from the multimedia controller, then perform the entry recording function as usual (s161). The process of checking any changes in the status of network thus finishes.

(0137)

In a case when any current multimedia device is disconnected from the repeater, the repeater operates in a manner described by reference to Fig. 49 below.

(0138)

The repeater examines the outputs of the node detecting circuits which correspond in one to one basis to its own ports (s188) and checks to see if any of the multimedia devices connected to the respective ports is

disconnected (s189). If it determines that a disconnection has occurred, then send a message for informing of the disconnection to the multimedia controller (s190) and finish its own service.

Meanwhile, the multimedia controller operates as follows. Referring to Fig. 47, it first examines whether or not there is an event either from the user or by communication (s162). If so, then examine whether or not that event is a message of the aforesaid disconnection (s163). If this results in determining that it is the disconnection message, then clear a "retry" counter (s169) and then clear the acknowledgment counter (s170). A message which is similar to the message that was used in the first embodiment for the purpose of detecting any change in the state of link to that multimedia device which is assumed to have been disconnected (s171). The aforesaid acknowledgment counter is used to measure a time-out period within which an answer to the detection message should be sent from the multimedia device in question to the multimedia controller. The aforesaid retry counter is used to determine a number of times the retry should be repeated as no acknowledgment comes back within the time-out period. Then, an acknowledgment to the first occurrence of the aforesaid detection message is waited for (s172). Upon advent of the acknowledgment, the multimedia device in question is determined to be still connected. The next event is then waited for. If no acknowledgment

arrives, then increment the acknowledgment counter (s173) and test if the time-out period has expired (s174). If not, return to s172 occurs and an acknowledgment is waited for. If the time-out period has expired, then increment the retry counter (s175) and test if the number of times the retry has repeated itself reaches a predetermined value (s176). It not yet, then return to the s170 and send the message for detecting any change in the status of route again.

(0139)

When the number of times the retry has been repeated just exceeds a predetermined value sensed by the retry counter, as this implies that no acknowledgment is given from the multimedia device in question, that route is determined to no longer exist. So the corresponding entry to this route is deleted from the network status monitoring table (s177). The table is thus updated.

(0140)

For another case, if the s163 determines that the event is not the message informing of having the aforesaid route removed, then examine whether or not it is the message informing of the continuance of the connection (s164). If so, then inspect if the multimedia device in question is already recorded in the network status monitoring table (s165). If its entry is found out therein, this table is left not rewritten. If, as not recorded yet, it is determined to be a new one, a

new entry is created in the network status monitoring table. The table is thus updated.

(0141)

Using the thus updated table, the aforesaid device IDs are redistributed over the current routes (s178), and the status of individual routes is displayed by respective icons for the multimedia devices (s179). To change the forms of these icons in response to change of the status of routes, and perform the operation that follows the actuation of the power on/off button, similar means and effects to those in the first embodiment of the invention may be used and result.

(0142)

A fourth embodiment of the invention is described below. Fig. 46 is a block diagram of the construction of this embodiment. As will be seen from Fig. 46, the fourth embodiment is fundamentally similar to the first one, but has characteristic features that the multimedia controller 1 does not contain the timer circuit 105 as was necessary in the first embodiment, that each of the multimedia devices is provided with at least a pair of node terminals and node detecting circuits ((183-4 and 183-5), (183-6 and 183-7) and (183-8 and 183-9)), which are similar to those described before, arranged in one to one correspondence with the node terminals, and that, as shown in Fig. 46, all the multimedia devices are connected in a chain configuration. In other words, it takes a form that an equivalent portion

to the repeater of the third embodiment shown in Fig. 45 is included in each of the multimedia devices. The operation of this network is described below.

(0143)

As an example, let us consider a case of Fig. 46 where a multimedia device C is connected to the multimedia device B. In this case, the multimedia device B plays a role of the repeater mentioned in the third embodiment of the invention. On connection of the multimedia device C, the node detecting circuit 183-7 of the multimedia device B sends the message informing of the establishment of a new connection to the multimedia controller.

(0144)

Let us consider another case where the multimedia device C of Fig. 46 is disconnected from the multimedia device B. In this case, too, the multimedia device B plays a role of the repeater of the third embodiment of the invention. On disconnection of the multimedia device C, the aforesaid node detecting circuit 183-7 of the multimedia device B sends a message informing of the removal of one of the current connections to the multimedia controller.

(145)

The routine for transferring the connection informing message from the multimedia device B is in a flowchart of Fig. 48, and the routine for transferring the disconnection informing message is in a flowchart

of Fig. 49. The multimedia controller processes these messages in the routine of Fig. 47. These routines are similar to those in the third embodiment of the invention.

(0146)

A fifth embodiment of the invention is next described. The construction of this embodiment is shown in Fig. 52. As will be seen from Fig. 52, the fifth embodiment is fundamentally similar in construction to the first embodiment of the invention, but has characteristic features that the multimedia controller does not contain the timer circuit 105 as was necessary in the first embodiment, that the multimedia devices each are provided with a node detecting circuits 138-10, 138-11 or 138-12 and a cable connector having a lock mechanism with a switch arranged to turn on when a release button is pushed so that a route cable becomes able to be pulled out. The aforesaid node detecting circuit is similar to that described in the third and fourth embodiments of the invention. The connector and the release button are shown in Fig. 53. When the free end of the route cable is inserted into the connector, the clock mechanism works so that the cable cannot be pulled out as it stands. To render this cable removable, the user has to push the release button. Just before the cable is pulled out, an actuating signal is applied to the aforesaid node detecting circuit, informing that the route is about to break.

(0147)

On receipt of this, the node detecting circuit sends a message informing of the removal of one of the current routes to the multimedia controller as in the third and fourth embodiments of the invention.

In another case that, as a cable is inserted, a new route is added to the network, the aforesaid node detecting circuit sends a message informing of the addition of a new route to the multimedia controller.

(0148)

As an example, let us consider the case of Fig. 52 on assumption that the multimedia device C is now connected in addition to the multimedia device B. In this case, having detected when the multimedia device C is connected to the communication line, its node detecting circuit transfers the message informing of the addition of a new route to the multimedia controller.

(0149)

Let us consider another case where to remove the route of from the multimedia device B to the multimedia device C. In this case, the user pushes the release button to make free the cable and then pulls the cable out of the connector. During this time, prior to the detachment of the cable, a signal representing the occurrence of removal of the cable is given off to the node detecting circuit 183-12. In turn, this circuit 183-12 transfers the message informing of the removal of one of the current routes to the multimedia controller.

(0150)

In such a manner, any changes in the status of individual routes are detected to make up a network status monitoring table, thus keeping track of all current multimedia devices. For the route removal message to send, the routine of Fig. 49 is used. For the route addition message to transfer, the routine of Fig. 48 is used. For the multimedia controller to operate, the routine of Fig. 47 is used. That is, these functions are executed similarly to the third and fourth embodiments of the invention.

(0151)

A sixth embodiment of the invention is next described. A characteristic feature of this embodiment is that, as shown in Fig. 50, all multimedia devices each are provided with a display section for a given identification name. No matter which of the first to fifth embodiments of the invention may be adopted, to keep track of all current multimedia devices, a network status monitoring table is made up in any way whatever. At a time when a new connection is established, or when one of the current connections is removed, the assignment of unique identification names to the current multimedia devices is updated as described in connection with the first embodiment of the invention. It is in the present embodiment that provisions are made for transferring these unique identification names to the respective individual multimedia devices and for displaying them in the

identification name display sections thereof.

(0152)

Taking an example of the system of Fig. 30 described in the first embodiment of the invention, the operation of the sixth embodiment of the invention is described below.

(0153)

Fig. 51 is a flowchart of the the main event loop for the multimedia controller 1. To begin with, examine whether or not there are any events as the user who is managing the multimedia controller has commanded, or as transmitted from the multimedia devices over the communication path (s180). If so, then execute its function (s182). If there is no more event, test if the "g" counter has established check time = 0 (s181). This counter determines a period in which the multimedia controller recycles the operation of checking the status of the network and is driven to count by the aforesaid timer circuit 105. If the s181 determines that check time = 0 is established, then reflect the detected changes in the status of connections to make up a network status monitoring table (s183) and assign device IDs to the multimedia devices recorded in this table.

(0154)

Based on the thus made table, rewrite the display of the status of individual links (s185) and set a value named "PreSet" to the "g" counter (s186). Also transfer the identification names recorded in the entries

in the "device name" column of the table shown in Fig. 33 to the respective individual multimedia devices (s187). For example, in the case of Fig. 33, the network contains two digital VTRs identified from each other by respective names: "Digital VTR 1" and "Digital VTR 2" as recorded in the corresponding entries in the "device name" column. Therefore, these names are transferred to the respective VTRs where they are displayed in the identification name display sections.

(0155)

Though the foregoing has been described mainly in connection with the construction of the first embodiment of the invention, it is to be understood that the principle of this sixth embodiment is applicable to any other cases of the second, third, fourth and fifth embodiments of the invention.

(0156)

(Advantages of the Invention)

If the system is formed based on the features described above, the user is no longer annoyed for carefully memorizing the status of all current connections. Even when a new device is added to the system, or when any one of the current devices is disconnected therefrom, the user is not required to do troublesome labour at all, because it has become possible to update the entries in the network status monitoring table while permitting the system to remain in the rising up state.

(0157)

The power sources of all the multimedia devices are made to manage and control unitarily, thereby giving another advantage that it becomes also possible to turn on and off the power sources to save electric energy efficiently. Thus, the system can be maintained economically.

(0158)

Further, to identify the current multimedia devices from one another, the names whose assignment is determined suitably by the multimedia controller are displayed on the identification name display sections, thereby giving an additional advantage that the user can visually make decision of which one of the devices he or she wants to use. Hence it becomes possible to architect a control system for multimedia devices which is acceptable to the ordinary users of little especial skill and appreciation on the computer systems.

(Brief Description of the Drawings)

(Fig. 1)

A diagram showing one form of the logic network of a multimedia controller and multimedia devices.

(Fig. 2)

Diagrams showing physical configurations of connecting a multimedia controller to multimedia devices.

(Fig. 3)

A diagram of the internal structure of an object

embedded multimedia device.

(Fig. 4)

A diagram of the internal structure of an object embedded multimedia controller.

(Fig. 5)

A diagram of a hierarchy of systems in the multimedia controller.

(Fig. 6)

A diagram, of a hierarchy of systems in the multimedia device.

(Fig. 7)

A diagram of a multimedia controller and a multimedia device before they are connected to each other.

(Fig. 8)

A diagram of the multimedia device connected to a LAN.

(Fig. 9)

A diagram of the structure of a common class library.

(Fig. 10)

A diagram of the structure of an object.

(Fig. 11)

A diagram of the structure of a system director object.

(Fig. 12)

A diagram of the structure of a control panel section in a file for describing a deputy object.

(Fig. 13)

A diagram of the structure of a data input/output object section in the file for describing a deputy object.

(Fig. 14)

A diagram of an object embedded digital VTR before connected to the multimedia controller.

(Fig. 15)

A diagram of the structure of a VTR controller object.

(Fig. 16)

A flowchart of the operation that follows the connection of the digital VTR to the LAN.

(Fig. 17)

A plan view of a multimedia controller window on the screen.

(Fig. 18)

A diagram of the object embedded digital VTR connected as a multimedia device to the LAN.

(Fig. 19)

A plan view of an icon for the digital VTR.

(Fig. 20)

A plan view of a control panel window on the screen.

(Fig. 21)

A diagram, partly in plan view, to explain the correspondence of the classes to which the objects belong with the constituent elements of the digital VTR control panel object.

(Fig. 22)

A diagram to explain the production of a play button object.

(Fig. 23)

Flowcharts of the routines for activation of the icon for the digital VTR and for start of a playing operation.

(Fig. 24)

A plan view of a multimedia controller window on the screen after the user has selected the control mode in the icon display for the digital VTR.

(Fig. 25)

A diagram of the relationship between the structure of the deputy "data input to digital VTR" object and the data for describing the object.

(Fig. 26)

A diagram of the relationship between the structure of the deputy "data output from digital VTR" object and the data for describing the object.

(Fig. 27)

A diagram of the structure of the "data input to digital VTR" object.

(Fig. 28)

A diagram of the structure of the "data output from digital VTR" object.

(Fig. 29)

A diagram of the relationship between the structure of the digital VTR control panel object of panel

class and the data for describing the object.

(Fig. 30)

A block diagram of a first embodiment of the invention.

(Fig. 31)

A block diagram of the structure of a digital VTR as a practical example of one of the multimedia devices appearing in the first, second, third, fourth, fifth and sixth embodiments of the invention.

(Fig. 32)

A flowchart of the event loop for the multimedia controller of the sixth embodiment.

(Fig. 33)

A schematic diagram of a network status monitoring table in the first, third, fourth, fifth and sixth embodiment of the invention.

(Fig. 34)

A flowchart of the routine for the operation of checking the status of the network in the first embodiment of the invention.

(Fig. 35)

Plan views expressing all current multimedia devices in windows on the screen according to the embodiments of the invention.

(Fig. 36)

A flowchart of the routine for the power on/off operation in the controller of the invention.

(Fig. 37)

A flowchart of the routine for the power on/off operation in the terminal device of the invention.

(Fig. 38)

A flowchart of the routine for the operation of the controller when an activating command is issued.

(Fig. 39)

A flowchart of th routine for the operation of the terminal device when an activating command is issued.

(Fig. 40)

A flowchart of the routine for controlling the event loop of each terminal device in response to actuation of the power on/off button.

(Fig. 41)

A block diagram of the construction of a second embodiment of the invention.

(Fig. 42)

A schematic diagram of the network status monitoring table in the second embodiment of the invention.

(Fig. 43)

A flowchart of the event loop for the multimedia controller in the second embodiment of the invention.

(Fig. 44)

A flowchart of the event loop for the multimedia device in the second embodiment of the invention.

(Fig. 45)

A block diagram of the construction of the

third embodiment of the invention.

(Fig. 46)

A block diagram of the construction of a fourth embodiment of the invention.

(Fig. 47)

A flowchart of the routine for the operation of updating the table in the third, fourth, fifth and sixth embodiments of the invention.

(Fig. 48)

A flowchart of the routine for the operation of detecting addition of a new connection in the flowchart of Fig. 47.

(Fig. 49)

A flowchart of the routine for the operation of detecting removal of one of the current connection in the flowchart of Fig. 47.

(Fig. 50)

A plan view of the outer appearance of the display in the multimedia device in the sixth embodiment of the invention.

(Fig. 51)

A flowchart of the event loop for the multimedia controller in the sixth embodiment of the invention.

(Fig. 52)

A block diagram of the construction of a fifth embodiment of the invention.

(Fig. 53)

A plan view of the outer appearance of a cable

connector in the fifth embodiment of the invention.

(Explanation of Reference Numerals)

1 multimedia controller;

102, 103, 104 multimedia Devices;

104 memory storing the network status monitoring table;

105 timer circuit in multimedia controller

106 system controller

107 interface controller

108 main power circuit

109 subsidiary power circuit

110 network interface unit

111 function unit in multimedia device

112 cable for route in network

113 analog output terminal

114 analog input terminal

115 D/A converter

116 A/D converter

117 memory for storing images

118 signal processing circuit

119 encoding circuit

120 decoding circuit

121 tape interface controller

122 tape deck unit

123 high speed data bus

124 control bus

125 display device in multimedia controller

126 network status monitor panel

24:04:95

127 selection panel for other applications

128 icon for database instrument

129 icon for CD player

130 icon for digital VTR

131 icon for digital camera

(Name of Document)

Drawings

(Fig. 1)

Logic of Connection of Multimedia Controller with multimedia devices

1: Multimedia Controller

2: Multimedia Device (Digital Camera)
2: Multimedia Device (Printer)
2: Multimedia Device (Digital FAX)
2: Multimedia Device (Digital Copier)
2: Multimedia Device (Digital VTR)
2: Multimedia Device (CD Player)

(Fig. 2)

Physical configurations of connecting Multimedia Controller to Multimedia Devices

a) Daisy Chain Line
   1: Multimedia Controller
   2: Multimedia Device

b) Star Configuration
   1: Multimedia Controller
   2: Multimedia Device

c) Multipoint Line
   1: Multimedia Controller
   2: Multimedia Device

(Fig. 3)

Internal Structure of Object Embedded Multimedia Device

10: external Bus
15: Multimedia Data
16: Driver for Mechanical System
17: Mechanisms & Motors
18: Driver for Electrical System
19: Electrical Circuits, Indicators & Switches

24.04.95

(Fig. 4)

Internal Structure of Object Embedded Multimedia Controller

25: Multimedia Data Filing
26: Display Controller
27: Display
28: Driver for Electrical System
29: Electrical Circuits, Indicators & Switches

(Fig. 5)

Hierarchy of Systems in Multimedia Controller

50: Hardware
53: Common Class Library
54: C Function
55: Specific Class Library
56: Application for Controlling Multimedia Devices

(Fig. 6)

Hierarchy of Systems in Multimedia Device

57: Hardware
59: Specific Class Library
60: C Function
61: Application for Multimedia Device

(Fig. 7)

1: Multimedia Controller
    205: System Director Object

2: Multimedia Device
    1061: File for Describing Deputy Multimedia Device Object
        1062: Section for Describing "Control Panel for Multimedia Device" Object
        1063: Section for Describing Deputy "Data I/O of Multimedia Device" Object
    1064: Multimedia Device Object
        1065: "Controller in Multimedia Device" Object
        1066: "Data Input to Multimedia Device" Object
        1067: "Data Output from Multimedia Device" Object

(Fig. 8)

1: Multimedia Controller
- 205: System Director Object
- 1068: Deputy Multimedia Device Object
  - 1069: "Control Panel for Multimedia Device" Object
  - 1070: Deputy "Data Input to Multimedia device" Object
  - 1071: Deputy "Data Output from Multimedia Device" Object

2: Multimedia Device
- 1064: Multimedia Device Object
  - 1065: "Controller in Multimedia Device" Object
  - 1066: "Data Input to Multimedia Device" Object
  - 1067: "Data Output from Multimedia Device" Object

(Fig. 9)

1081: Class Library
- 1079: First Class
  - 1080: Class Definition Part
  - 1081: Class Method Table
  - 1082: Code Part
    - 1083: 1st Function Code
    - 1084: k-th Function Code
- 1085: p-th Class

(Fig. 10)

234: Object
- 235: Portion for Internal Data
  - 236: 1st Internal Data
  - 237: 2nd Internal Data
  - 238: n-th Internal Data
- 239: Portion for Methods
  - 240: 1st Data Processing Means
  - 241: 2nd Data Processing Means
  - 242: m-th Data Processing Means
- 243: Class Method Table
  - 240: 1st Data Processing Means
  - 241: 2nd Data Processing Means
  - 242: 3rd Data Processing Means
- 244: Portion for Accommodating Pointer for Class Method Table
- 245: Message Communicating Means; Message
- 246: Processing & Retrieving Means

External Data

(Fig. 11)

205: System Director Object
- 342: Processing & Retrieving Means
- 1061: File for Describing Deputy Multimedia Device Object
- 1072: Portion for Accommodating Pointer for Class Method Table
- 1073: Class Method Table for System Director
  - 1047: Deputy Multimedia Device Object Forming means
  - 343: Data I/O Coordinating Means
  - 380: Application Object Forming Means
- 1074: Message Communicating Means; Message
- 1075: Portion for Methods
  - 1047: Deputy Multimedia Device Object Forming means
  - 343: Data I/O Coordinating Means
  - 380: Application Object Forming Means
- 1076: Portion for Internal Data
  - 1077: Object ID
  - 1078: Object Registration Data
  - 344: Control Data for linkage between Devices

(Fig. 12)

247: Section for Describing Control Panel Object
- 248: 1st Database for Describing Object
  - 250: Data for Recognizing Object
    - 251: Class Name
    - 252: Object ID
    - 253: Superior Object ID
  - 254: Data for Drawing Object
    - 255: 1st Data for Drawing Object
      - 256: Data for Location & Size
      - 257: Data for Pattern & Color
      - 258: Object Image
    - 259: j-th Data for Drawing Object
  - 260: Data for Object Link
    - 261: 1st Data for Object Link
      - 262: Relational Object ID
      - 263: Message for Transmission
    - 264: k-th Data for Object Link
- 249: i-th Database for Describing Object

(Fig. 13)

650: File for Describing Deputy Data I/O Object
- 651: 1st Data for Deputy Data Input Object
  - 652: Object ID
  - 653: Relational Data Input Object ID
  - 654: Compatible File Type List
- 655: m-th Data for Deputy Data Input Object
- 659: 1st Data for Deputy Data Output Object
  - 660: Object ID
  - 661: Relational Data Output Object ID
  - 662: Compatible File Type List
- 663: n-th Data for Deputy Data Output Object

(Fig. 14)

1: Multimedia Controller
- 205: System Director Object

203: Digital VTR
- 206: Digital VTR Object
  - 207: Digital VTR Controller Object
  - 208: "Data Input to Digital VTR" Object
  - 209: "Data Output from Digital VTR" Object
- 210: File for Describing Deputy Digital VTR Object
  - 211: Section for Describing "Control Panel for Digital VTR" Object
  - 212: Section for Describing Deputy "Data Output from Digital VTR" Object

(Fig. 15)

207: Digital VTR Controller Object
- 1009: Portion for Accommodating Pointer for Class Method Table
- 1010: Message Communicating Means
- 1011: Processing & Retrieving Means
- 1012: Portion for Methods
  - 1019: Reproducing Means
  - 1020: Recording Means
- 1015: Portion for Internal Data
  - 204: Object ID
  - 1016: Tape Running Speed
  - 1017: Current Footage
- 1018: Class Method Table for Controller Class
  - 1019: Reproducing Means
  - 1020: Recording Means

(Fig. 16)

636: Connect Digital VTR to Network
637: System Director Object Detects When Digital VTR is Connected
638: System Director Object issues Device ID to Digital VTR
639: System Director Object Loads File for describing Deputy Digital VTR Object
640: System Director Object Generates Deputy Digital VTR Object Based on Information from the File
641: Deputy Digital VTR Object Displays Icon (Mini Panel) for Digital VTR in System Controller Window
642: Wait for Instruction from User

(Fig. 17)

(Fig. 18)

1: Multimedia Controller
- 205: System Director Object
- 220: Deputy Digital VTR Object
  - 221: Deputy "Control Panel for Digital VTR" Object
  - 222: Deputy "Data Input to Digital VTR" Object
  - 223: Deputy "Data Output from Digital VTR" Object

203: Digital VTR
- 206: Digital VTR Object
  - 207: Digital VTR Controller Object
  - 208: "Data Input to Digital VTR" Object
  - 209: "Data Output from Digital VTR" Object

(Fig. 19)

(Fig. 20)

(Fig. 21)

221: Panel Class: Control Panel for Digital VTR
- 285: Menu Class: Panel View Option Menu
- 286: Form Class: Timer Counter
- 288: Button Class: Rewind Button
- 289: Button Class: Reverse Play Button
- 290: Button Class: Pause Button
- 291: Button Class: Fast Feed Button

292: Button Class: Stop Button
293: Button Class: Recording Button
294: Button Group Class: Selection of Control Modes
295: Radio Button Class: Default Button
296: Radio Button Class: Advanced Button

(Fig. 22)

612: Play Button Object
613: Portion for Accommodating Pointer for Class Method Table
614: Message Communicating Means
615: Processing & Retrieving Means
616: Portion for Methods
626: Button Initializing Means
627: Button Drawing Means
628: Click Response Means
620: Portion for Internal Data
621: Object ID
622: Data for Button State
623: Drawing Parameter
624: Link Data
625: Button Class Method Table for Button Class
626: Button Initializing Means
627: Button Drawing Means
628: Click Response Means
625: Button When Not Pressed
626: Button When Pressed
297: Data for Object Recognition
298: Class Name: Button Class
299: Object ID: ID=7 Play Button
300: Super Object ID; ID=1 VTR Control Panel
601: 1st Data for Drawing Object
602: Data for Location & Size
603: Data for Pattern & Color
604: Object Image
605: 2nd Data for Drawing Object
606: Data for Location & Size
607: Data for Pattern & Color
608: Object Image
609: Object Link Data
610: Link Terminal Object ID ; ID of VTR Control Object
6121: Message to Transmit; Play

(Fig. 23)

643: User Double Clicks on Icon of Digital VTR
644: "Control Panel for Digital VTR" Object Presents Display of Control Panel for Digital VTR
645: Wait for User's Action

646: Click on Play button
647: "Control Panel for Digital VTR" Object Sends "Play" Message to Digital VTR Controller Object
648: Digital VTR Controller Object Activates Reproducing Means
649: Start Play Mode of Digital VTR

(Fig. 24)

(Fig. 25)

222: Deputy "Data Input to Digital VTR" Object
- 668: Portion for Accommodating Pointer for Class Method Table
- 669: Message Communicating Means
- 670: Processing & Retrieving Means
- 671: Portion for Methods
  - 680: Means for Initializing Deputy Data Input Object
  - 681: Link Data Updating Means
  - 678: Compatible File Type Replying Means
- 674: Portion for Internal Data
  - 675: Object ID
  - 676: Relational Data Input Object ID
  - 677: Compatible File Types
  - 1006: Link Data
- 679: Class Method Table For Deputy Data Input Class
  - 680: Means for Initializing Deputy Data Input Object
  - 681: Link Data Updating Means
  - 678: Compatible File Type Replying Means
- 682: Deputy Data Input Object Data
  - 683: Object ID
  - 684: Relational Data Input Object ID
  - 685: Compatible File Type List

(Fig. 26)

223: Deputy "Data Output from Digital VTR" Object
- 690: Portion for Accommodating Pointer for Class Method Table
- 691: Message Communicating Means
- 692: Processing & Retrieving Means
- 693: Portion for Methods
  - 694: Means for Initializing Deputy Data Input Object
  - 695: Means for Updating Link Data
  - 700: Means for replying Compatible File Type
- 696: Portion for Internal Data
  - 697: Object ID
  - 698: Relational Data Output Object ID
  - 699: Compatible File Types

688: Link Data
1048: Class Method Table for Deputy Data Output Class
694: Means for Initializing Deputy Data Output Object
695: Means for Sending Data Input Command
700: Means for replying Compatible File Type
1001: Deputy Output Object Data
1002: Object ID
1003: Relational Data Output Object ID
1004: Compatible File Type List

(Fig. 27)

208: "Data Input to Digital VTR" Object
1022: Portion for Accommodating Pointer for Class Method Table
1023: Message Communicating Means
1024: Processing & Retrieving Means
1025: Portion for Methods
1032: Means for Writing File
1033: Means for Receiving Data
686: Means for updating Link data
1028: Portion for Internal Data
1029: Object ID
1030: Link Data
1031: Class Method Table for Data Input Class
1032: Means for Writing File
1033: Means for Receiving Data
686: Means for Updating Link Data

(Fig. 28)

209: "Data Output from Digital VTR" Object
1035: Portion for Accommodating Pointer for Class Method Table
1036: Message Communicating Means
1037: Processing & Retrieving Means
1038: Portion for Internal Data
1045: Means for reading Data
1046: Means for Transmitting Data
687: Means for Updating Link Data
1041: Portion for Internal Data
1042: Object ID
1043: Link Data
1044: Class Method Table for Data Output Class
1045: Means for Reading Data
1046: Means for Transmitting Data
687: Means for Updating Link Data

(Fig. 29)

221: "Control Panel for Digital VTR" Object
- 1401: Portion for Accommodating Pointer for Class Method Table
- 1402: Class Method Table for Panel Class
  - 1403: Means for initializing Panel
  - 1404: Means for Drawing Panel Drawing
  - 1405: Click Response Means
- 1408: Portion for Methods
  - 1403: Means for Initializing Panel
  - 1404: Means for Drawing Panel Drawing
  - 1405: Click Response Means
- 1410: Portion for Internal Data
  - 1411: Object ID
  - 1412: Data for Panel State
  - 1413: Drawing Parameters
- 1414: Data for Object Recognition
  - 1415: Class Name; Button Class
  - 1416: Object ID; ID = 1 Digital VTR Control Panel
  - 1417: Super Object ID
- 1418: 1st Data for Drawing Object
  - 1419: Data for Location & Size
  - 1420: Data for Pattern & Color
  - 1421: Object Image
- 1422: 2nd Data for Drawing Object
  - 1423: Data for Location & Size
  - 1424: Data for Pattern & Color
  - 1425: Object Image
- 1426: Icon Image
- 1427: Control Panel for Digital VTR (Frame)

(Fig. 30)

1: Multimedia Controller
- 103: Control Function Unit
- 104: Memory storing Monitoring Table
- 105: Timer Circuit
- 106-0: System Controller
- 107-0: Interface Controller
- 108-0: Power Circuit 1
- 109-0: Power Circuit 2

102-1: Multimedia Device A:
- 106-1: System Controller
- 107-1: Interface Controller
- 108-1: Power Circuit 1
- 109-1: Power Circuit 2
- 110-1: Network Interface Unit
- 111-1: Multimedia Function Unit

(Fig. 31)

106: System Controller
107: Network Interface Controller
108: Power Circuit 1
109: Power Circuit 2
112: Network Line
113: Analog output
114: Analog Input
117: Memory
118: Signal Processing Circuit
119: Encoding Circuit
120: Decoding Circuit
121: Tape Interface Controller
122: Tape Deck Unit

(Fig. 32)

Event Loop for Multimedia Controller
S101: Is There Any Event ?
S102:
S103: Check Status of Network
S104: Distribute Device ID
S105: Display Status of Network
S106:
S107: Process Event

Event: There are two forms of events, one of which occurs when the user manipulates the multimedia controller and the other of which is transmitted from the terminal device by communication.

g Check Time: The time for which the counter "g" has operated as the multimedia controller is checking the status of the network. The counter is decremented by an external timer or a software interrupt timer.

Pre Set: The value of the period in which the multimedia controller recycles the operation of checking the status of the network

(Fig. 33)

Line Check; Power On/Off; Device ID; Article Name; Unique Number; Address of Location storing Object Data

(Fig. 34)

Check Status of Network
S107:
S108: Issue Message for Detecting Change in Status of Network
S109:
S110
S111: Is Reply Message Received ?
S112: Inspect Network Status Monitoring Table to determine whether or not the device in question is already known
S113: Is It Known ?
S114: Add Entry into Table
S115: Check Corresponding Entry to device in Table.
S116:
S117: Are there any more in Table?
S118:
S119: Issue Message for Detecting Change in the status of Network to Device not checked in Table
S120:
S121
S122: Is Reply Message Received ?
S123: Is Status of Link OK ?
S124; Check corresponding Entry in Table
S125: delete Corresponding Entry out of table
End

(Fig. 35)
Digital VTR2

(Fig. 36)
Power On/Off
S126: Issue Message for Turning on/off Main Power Source to selected device.
S127:
S128: Is There Reply ?
S129:
S130:
S131: Is Start OK ?
S132: Issue Error Message
S133: Update Entry in power on/off column in Network Status Monitoring Table
End; Controller

(Fig. 37)
Power On/Off
S134: Start Operation of Turning on Main Power Source
S135: Has Start completed Normal ?
S136: Send Message of Normal Start Completion
S137: Send Message of Abnormal Start Interrupt
End; Terminal Device

(Fig. 38)

Activation Command
S190: Issue Activation command to Selected Multimedia Device
S191:
S192: Is there Reply ?
S193:
S194:
S195: Is Start OK ?
S196: Issue Error Message
S197: Update Entry in power on/off column in Network Status Monitoring Table
S198: Update Entry in power on/off column in Table
End; Controller

(Fig. 39)

Activation Command to Receive
S199: Is Main Power Source now ON ?
S500: Turn on Main Power Source
S501: Is Start Normal ?
S502: Execute Command
S503: Is Operation Normal ?
S504: Issue Message of Normal Start Completion
S505: Turn Off Main Power source
S506: Issue Message of abnormal start Interrupt
End; Terminal Device

(Fig. 40)

Event Loop for Every Device
S507: Is There Any Event ?
S508:
S509:
S510: Turn off Main Power Source
S511: Issue Message of Having Main Power Source Turned off
S512:
S513: Process Event
S514:
S515:

(Fig. 41)

1: Multimedia Controller
103: Control Function Unit
104: Memory Storing Monitoring Table
105: Timer circuit 2
106-0: System Controller
107-0: Interface Controller
108-0: Power Circuit 1
109-0: Power Circuit 2

102-1: Multimedia Device A
106-1: System Controller
107-1: Interface controller
108-1: Power circuit 1
109-1: Power Circuit 2
110-1: Network Interface Unit
111-1: Multimedia Function Unit
136-1: Timer Circuit 1

(Fig. 42)

Link Check; Power on/off; Device ID; Article Name; Unique Number; Address of location storing Object Data

(Fig. 43)

Event Loop for Multimedia Controller
S138: Is There Any Event ?
S139:
S140: Delete Corresponding Entry out of table
S141: Link Status Message ?
S142: Inspect Network Status Monitoring Table
S143: Process Event
S144: Already Recorded in Table ?
S145: Create New Entry in Table
S146:
S147: Display Status of Network

(Fig. 44)

Event Loop for Every Device
S148: Is There Any Event ?
S149: Process Event
S150:
S151: Issue Connection Message
S152:

Event: There are two forms of events, one of which occurs when the user manipulates the device, and the other of which is transmitted from the multimedia controller by communication.

f Refresh Counter:
Determine a period in which the device recycles the operation of informing its own state of connection to the multimedia controller. The counter is decremented by an external timer, or a software interrupt timer.

Pre Set: The Value of the period in which the device recycles the operation of informing its own state of connection to the multimedia controller

(Fig. 45)

1: Multimedia Controller
- 103: Control Function Unit
- 104: Memory Storing Monitoring Table
- 106-0: System Controller
- 107-0: Interface Controller
- 108-0: Power Circuit 1
- 109-0: Power Circuit 2

102-1: Multimedia Device A
- 106-1: System controller
- 107-1: Interface Controller
- 108-1: Power Circuit 1
- 109-1: Power circuit 2
- 110-1: Network Interface Unit
- 111-1: Multimedia Function Unit

137: Repeater
- 107-4: Interface Controller
- 138-1; 138-2; 138-3: Node Detecting Circuit
- 138-4: Power Circuit
- 139: Control Unit

(Fig. 46)

1: Multimedia Controller
- 103: Control Function Unit
- 104: Memory Storing Monitoring Table
- 106-0: System Controller
- 107-0: Interface Controller
- 108-0: Power Circuit 1
- 109-0: Power Circuit 2

102-1: Multimedia Device A
- 106-1: System Controller
- 107-1: Interface Controller
- 108-1: Power Circuit 1;
- 109-1: Power Circuit 2
- 110-1: Network Interface Unit
- 111-1: Multimedia Function Unit
- 138-4; 138-5: Node Detecting circuit

(Fig. 47)

Event Loop for Multimedia Controller
S162: Is There Any Event ?
S163: Removal Message ?
S164: Addition Message ?
S165: Inspect Network Status Monitoring Table
S166: Already Recorded in Table ?
S167: Record New Entry in Table
S168: Process Event

S169:
S170:
S171: Issue addition Message
S172: Is ACK Received ?
S173:
S174:
S175:
S176:
S177: Delete Corresponding Entry out of Table
S178: Distribute Device ID
S179: Display Status of Network

(Fig. 48)

Route Addition
S153: Test Output of Node Detecting circuit
S154: Is Node connected ?
S155: Issue Addition Message
S156:
S157: Is ACK Received ?
S158:
S159:
S160: Process Error
S161: Detect Addition
End

(Fig. 49)

Route Removal
S188: Test Output of Node Detecting Circuit
S189: Removed ?
S190: Issue Removal Message
End

(Fig. 50)

141: Display Section for Device Identification Name;
Digital VTR1

(Fig. 51)

Event Loop for Multimedia Controller
S180: Is There Any Event ?
S181:
S182: Process Event
S183: Check Status of Links
S184: Distribute Device ID
S185: Display Status of Links
S186:
S187: Issue Device Identification Names

Event: There are two forms of events, one of which occurs when the user manipulates the multimedia controller

and the other of which is received from the terminal device by communication.

g Check Time:
Counter for determining the period in which the multimedia controller recycles the operation of detecting changes in the status of the network. The counter is decremented by an external timer or a software interrupt timer.

Pre Set: The value of the period in which the multimedia controller recycles the operation of detecting change in the status of the network.

(Fig. 52)

1: Multimedia Controller:
103: Control Function Unit
104: Memory Storing Monitoring Table
106-0: System Controller
107-0: Interface Controller
108-0: Power Circuit 1
109-0: Power Circuit 2

102-1: Multimedia Device A
106-1: System Controller
107-1: Interface Controller
108-1: Power circuit 1
109-1: Power circuit 2
110-1: Network Interface Unit
111-1: Multimedia Function Unit
138-10: Node Detecting circuit

(Fig. 53)

112: Route Cable
106: System Controller
107: Interface Controller
108: Power source
138: Node Detecting circuit
142: Release/Actuation Button for Lock Mechanism/Cable Pull-out Sensor Switch
143: Connector

24.04.95

(Name of Document)

Written Abstract

(Abstract)

(Object) To accurately grasp the status of links between a multimedia controller and each of multimedia devices without the user having to be conscious about it, thus improving the efficiency of the entirety of the system.

(Constitution) To provide for the multimedia device with a first control unit for communication with the multimedia controller; for the first control unit with an interface controller and a first system controller for communication with the multimedia device and for controlling it; for the multimedia controller with a second control unit for controlling communication with the multimedia devices; for the second control unit with means for processing signals informing of connection or disconnection of multimedia devices, means storing a network status monitoring table which is updated based on the signal processing means and a second system controller for communicating with the multimedia controller and for control.

(Selected Figure) Fig. 30